Final project literature review

Self-supervised pipeline for automated product replacement based on neural language models

Andrei Ionut Damian
XCS224U Spring 2020

Abstract

In past years we have seen a lot of cross-domain application from various deep learning areas to particular real-life cases with apparently little connection. One of those areas is that of applying deep representation learning based on neural language processing to business analytics system. This document presents the summarization and synthesis of several approaches in the area of Natural Language Understanding as well as several papers relevant to the current state of the art in the area of recommender systems based on neural language processing and representation learning in general. This cross-domain analysis is required to further define and test the proposed hypothesis of inferring product replacement retail and distribution systems. We will try to review the proposed natural language understanding papers as tools for our experiments, as well as define a potential outline of the envisioned representation learning pipeline for product replacement inference. The proposed range of analyzed research subjects will be range from word-embedding vector space generations, retrofitting methods for word embedding required to better capture semantic relationship up to analyzing two different views on the problem of product recommender systems.

1 Introduction and problem definition

Deep representation learning is arguably one of the most important areas of machine learning as it allows us to automatically – and potentially in un-supervised fashion – discover rich features of the raw data. Since the introduction of the well-known word-embeddings generation algorithms in early 2015, the area on natural language processing and understanding has seen a tremendous improvement. Although it might seem that the actual state-of-the-art resides strongly on contextual embeddings and/or complex multi-head self-attention architectures, it is all based on the initial "basic" steps in the area of semantic vector space models.

Based on this rich area of research and experimentation a lot of development has been done in the area of applying NLP/NLU approaches to commercial transaction systems including recommender systems. One the most common approaches, further described later in section 2, is to view a retail transaction (a list of purchased items) as a natural language object – a context window, the whole retail transactional database as the actual text corpus and the item SKU database as the "word vocabulary". This approach of creating a hypothesis similarity between the natural language representation learning and business analytics systems has proved successful in various cases and it is still a very active area of research.

1.1 Intuition and list of related papers

The intuition behind our proposed deep learning pipeline is that we could, at least in theory, generate, through multiple modeling iterations, powerful-enough semantic vector space embeddings for each individual item (product) so that we can infer replacements (synonym) items

and propose them in the case on original item shortages – all of these in a self-supervised setting. In order to summarize our proposed papers we have the following lists of review objectives:

1) Semantic vector space model generation approaches such as *GloVe, skip-gram, BOW* (*Word2Vect)*
2) Cross-domain application of NLU in models for product recommendation systems
3) Comparison with state-of-the-art in product recommendation systems based on classic approaches
4) Basic retrofitting techniques
5) Counter-fitting approaches
6) Advanced (functional) retrofitting
7) Other semantic vector space enrichment approaches

2 Papers summaries

In the following sub-sections, a deeper look will be given at various approaches and "tools" we plan to employ in our proposed experiment. We will analyze a mix of fundaments of deep representation learning including but not limited to semantic vector space model generation as well as representation learning based recommender systems. As our target is to clearly define our related concepts and approaches, we will present each approach in a general perspective and also pinpoint our the take-aways.

2.1 Semantic vector space models

One of the most important papers relevant to our subject is *"GloVe: Global Vectors for Word Representation"* (Pennington, Socher, & Manning, 2014) as it describes one the core representation learning methods required to generate sound semantic word embeddings beside the similarly well-known *word2vect* (Mikolov, Chen, Corrado, & Dean, 2013). The main intuition behind *GloVe* is based on the fact that statistics of word co-occurrence in a corpus is the primary source of information of the proposed unsupervised method for learning word representations. Thus, the proposed word vector generation pipeline first constructs the matrix of co-occurrence then applies a weighted least squares regression using as target the natural logarithm of the word counts as defined by the below objective function.

$$J = \sum_{i,j=1}^{V} f(X_{ij})\ (w_i^T \tilde{w}_j\ +\ b_i + \tilde{b}_j - logX_{ij})^2 \tag{1}$$

Finally, as the model generates through the optimization process to sets of word vectors (W and $\tilde{W}$), the *GloVe* algorithm summarizes the two matrices in order to obtain the final semantic vector space models.

Several other aspects of interest can be found in the paper for the construction of our hypothesis and

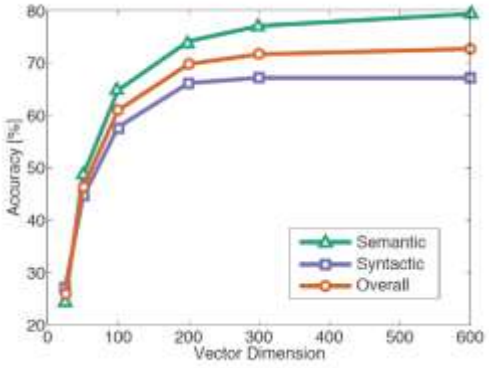

Figure 1 - Accuracy on a specific task as a function of vector size. Image from original paper by Pennington et al

one of the most important is the correlation between word vector dimension (and thus model capacity) and the task-related accuracy (see Figure 1).

As the intuition suggests increasing model capacity (bigger vector space dimension) allows capturing richer semantic knowledge that in turns leads to more useful higher-level features and better accuracy down the stream. This is another of the main take-aways for our proposed pipeline – varying and searching for the optimal product representation dimensionality.

2.2 Product recommendations

The logical next step in our related work review would be to analyze one of the early research papers that proposed the encoding of item SKUs in a similar manner to that of words. Grbovic

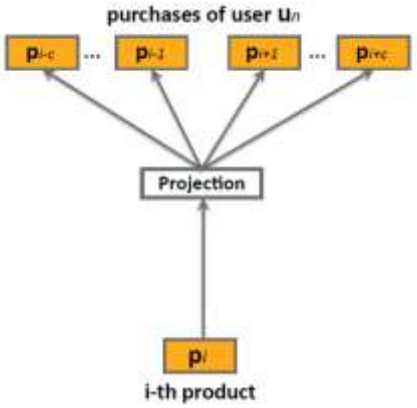

Figure 2 - The skip-gram architecture similar with original word2vec paper. Image from original paper by *Grbovic et al*

et al proposed in their paper "*E-commerce in your inbox: Product recommendations at scale*" (Grbovic, et al., 2015) a direct approach of constructing product embeddings based on word2vec skip-gram model.

The authors propose a direct analogy between a certain product (item) in a basket and the focal word in the skip-gram context as presented in Figure 2.

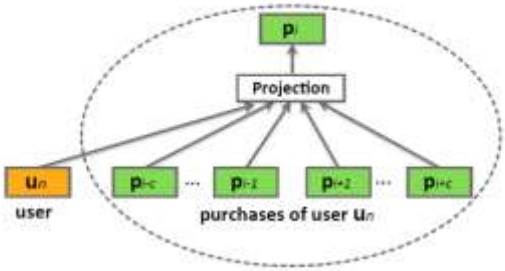

Figure 3 - Bag-of-words like approach to enrich the vector space model with user meta information. Picture based on the original paper by Grbovic et al

Another proposed aspect was that of enriching the vector space model with meta information such as user-identification in a similar method with *doc2vec* approach – a follow-up of the original word2vec – by Le and Mikolov (Le & Mikolov, 2014). In Figure 3 is presented this particular approach, this time being based on the BOW algorithm (predicting a focal item based on the context) instead of the skip-gram (predicting the context starting from a focal item).

In terms of actual vector space model applications, the authors propose two different hypotheses:

- straight correlation between items cosine distances and the actual real-life similarities of the products (namely the **prod2vec-topK** approach)

- a second more elaborate heuristic approach of finding viable "complementarity" baskets of products that might be sold well together. This second proposed predictive model, called **prod2vec-cluster** by the authors, leverages the hypothesis that similar items can be grouped together using distance metrics with K-Means algorithm. Furthermore, close clusters generated in this manner yield potentially well correlated products and picking items from different close clusters will generate a well-defined complementarity basket.

This particular work is one of the earliest and most well known in the area of applying representation learning and neural language modelling to the problem of retail recommender system. In contrast with this work, the ACM Recommender Systems Challenge'17 was won by a more recent yet more conservative approach by Volkovs et al. In their paper "*Content-based Neighbor Models for Cold Start in Recommender Systems*" (Volkovs, Yu, & Poutanen, 2017) the authors propose a classic approach based on heavy manual featurization of the input data and full supervised approach.

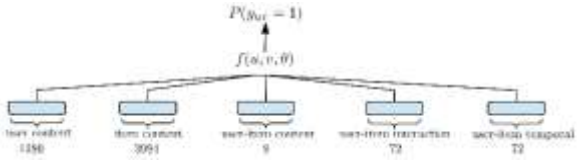

Figure 4 - Figure from paper of Volkovs et all presenting the overall architecture of their classification model using over 4000 hand-crafter features

Although the proposed approach won the competition, we believe that availability of data as well as relying on well annotated data and supervised datasets will become more and more inefficient for large scale systems deployment with good generalization capacity. As a final note, in this particular approach the model is not able to map unknown relationships or generate semantic knowledge out of unsupervised data which is one of our main goals.

For our proposed pipeline for products embeddings, we plan to employ *GloVe* approach to vector space generation as well as use the discounted positive pointwise mutual information taken from the matrix of item co-occurrence. In our further experiments we will try to demonstrate that applying *GloVe* approach will yield similar if not better results in obtaining basic semantic vector space representations of products.

2.3 Retrofitting and counter-fitting

Following the analysis of the base tool – the *GloVe* word-embeddings generation approach and its particular (and potential) cross-domain application to recommender systems - it is now required that we revisit research work in the area of enriching vector space models based on retrofitting and counter-fitting. For this subject we decided to analyze three important papers: "*Retrofitting Word Vectors to Semantic Lexicons*" by Faruqui et al (Faruqui, et al., 2014), "*Counter-fitting word vectors to linguistic constraints*" (Mrkšić, et al., 2016) and the more recent work of Benjamin J. Lengerich, Andrew L. Maas and Christopher Potts "*Retrofitting Distributional Embeddings to Knowledge Graphs with Functional Relations*" (Lengerich, Maas, & Potts, 2017).

Basic retrofitting

In order to better understand the motivation for the review of vector space models retrofitting approaches (and in a later section the review of the counter-fitting approaches) we have to revisit the work done by *Grbovic at al* as well as other teams that pursued the goal of obtaining products semantic vector space models. In all above approaches the authors base their work on the shallow hypothesis that similar items (*words*) measured by cosine distance might have a synonym-like or an entailing-like relationship. However, this as possible for product vector space models as it is with natural language vector space models counterparts. For example, a *GloVe-300* vector space model, pretrained on *Wikipedia* and *Gigaword* might tell us that *'adolescent'*, *'teenager'*, *'puberty'* are very similar to each other based on their pairwise cosine distance and actually '*adolescent'* is more similar to '*puberty'* than it is to '*teenager'*. Exactly the same stands with product semantic vector space models – just using a raw embedding generated with *GloVe* or *word2vect* is not enough to get rich and reliable properties of the items.

This is the point where adding meta-information similar to the *synonymity graphs* in natural language might greatly aid us in further refining the semantic power of our product vector space model. The main intuition is well defined in the work of Faruqui et all as "*graph-based learning technique for using lexical relational resources to obtain higher quality semantic vectors*" by means

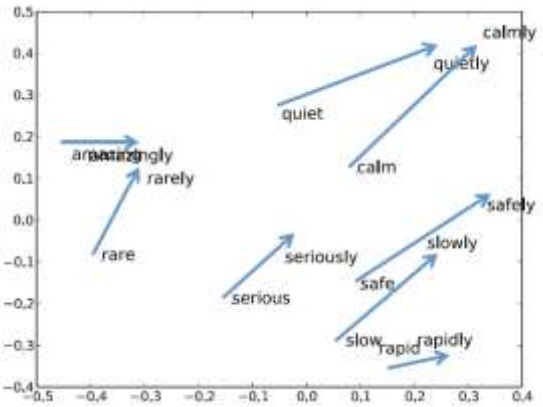

Figure 5 - Closing the distance between adjective-adverb pairs. Figure from original paper by Faruqui et al presenting the post-retrofit vector space projection

of post-processing the pre-trained semantic vector space models.

$$J = \sum_{i=1}^{N} \left[\alpha_i \|q_i - \hat{q_i}\|^2 + \sum_{i,j \in E} \beta_{ij} \|q_i - q_j\|^2 \right] \quad (2)$$

Another important aspect is that the proposed "*retrofitting*" optimization method, described by the objective function in equation 2, is agnostic to the origins of the input vector space model, such as original training objective or overall initial training approach. In the this equation Q is the new optimized vector space matrix, $\hat{Q}$ is the original vector space model, and *i,j* are pairs of words that are connected by an edge in *E.*

As the authors explain in the paper, they perform experiments using various semantic lexicons such as PPDB (Ganitkevitch, Van Durme, & Callison-Burch, 2013) and WordNet (Miller, 1995), use these in order to improve the word vectors. Following the optimization process they evaluate the quality of the new *retrofitted* word vectors in order to determine how well they capture semantic aspects.

In another paper on the subject of counter-fitting (Mrkšić, et al., 2016) the authors take a more generalized approach by deciding to intuitively pull together the synonymous words while pushing antonymous words vectors apart. The pushing operation proposed by Mrkšić et *al* can be viewed as a similar operation to that proposed by the work of Faruqui et al although the optimization process is slightly different. *The counter-fitting* method proposes a three-part objective function as follows:

- the first component, described below in equation (3), of the objective function is responsible of antonym pushing by imposing a minimal *distance* δ (using a distance function *d*) for the antonym word embeddings (τ is *max* function)

$$J_A = \sum_{u,w \in A} \tau\,(0, \delta - d(v'_u, v'_w)) \quad (3)$$

- the second term – equation (4) - purpose is to pull synonym vectors closer by reducing the distance between them to a certain margin and it is closely related with the second term from the objective function in the Faruqui et al work described by equation (2).

$$J_S = \sum_{u,w \in A} \tau\,(0, d(v'_u, v'_w)) \quad (4)$$

- the third and final component presented in equation (5) has the purpose of preserving the overall vector space structure by minimizing the difference between the original vector space word embedding pairs and the new *counter-fitted* word embeddings. This particular term is quite similar – and has the same purpose - with the first term in the objective function proposed by Faruqui et all (Faruqui, et al., 2014)

$$J_{VSP} = \sum_{i=1}^{n} \sum_{j \in N(i)} \tau\,(d(v'_u, v'_w) - d(v'_u, v'_w), 0) \quad (5)$$

Finally, the full objective function in equation of the *counter-fitting* approach is simply the weighted sum of the three components as follows:

$$J = w_1 * J_a + w_2 * J_s + w_3 * J_{VSP} \quad (6)$$

Short intuitive relationship to our proposed experiment

Intuitively we plan to employ the same approach of *retrofitting* and *counter-fitting* the *GloVe* generated **product vectors** with existing meta-information in the retail databases – information such as product categories or even detailed category management tree-structures. The hypothesis is that this approach will reduce the cosine distance between products that can actually replace each other in real life in a similar manner as presented in the work of Faruqui et al that addresses word vectors.

Advanced retrofitting

Recent work of Benjamin J. Lengerich, Andrew L. Maas and Christopher Potts show that some of the core assumptions of the original base *retrofitting* approach of Faruqui et al, such as that

connected entities should have similar embeddings, cannot hold for any real-life applications - such as health knowledge graphs. In order to address the underlined limitations, the authors propose *Functional Retrofitting* - a retrofitting approach that sees pairwise entity relations as functions rather than simple embedding straight similarities. One of the first implication of this approach is that we can include both the *pull* and the *push* between embeddings based on real-life similarity or dissimilarity in a more robust and generalized way that the one presented by *Mrkšić et al*. In order to further understand the approach proposed by the authors of the *Functional Retrofitting* let us understand the objective function in equation (7) by dissecting each of its four components. In order to simplify the explanations, we use a slightly different notation than in the original paper (Lengerich, Maas, & Potts, 2017)

$$J(Q,F) = J_{VSMP} + J_C + J_D + \sum_{r\in R} \rho_\lambda(f_r) \quad (7)$$

$$J_{VSMP} = \sum_{i\in Q} \alpha_i \|q_i - \hat{q}_i\|^2 \quad (8)$$

$$J_C = \sum_{(i,j,r)\in \mathcal{E}} \beta_{i,j,r} f_r(q_i, q_j) \quad (9)$$

$$J_D = \sum_{(i,j,r)\in \mathcal{E}^-} \beta_{i,j,r} f_r(q_i, q_j) \quad (10)$$

The first component J_{VSMP} of the objective function is the semantic vector space model preservation constraint, identical to that in (Faruqui, et al., 2014) and similar to equation (5) of (Mrkšić, et al., 2016). The second J_C and third J_D terms represent the application of relational penalty function for both the positive-relationship $\mathcal{E}$ knowledge-graph as well as for the $\mathcal{E}^-$ *negative space*. We can view equation (9) as the second term in the objective function from (Faruqui, et al., 2014) and respectively the J_S – equation (4) – from (Mrkšić, et al., 2016). Although equation (10) does not have an intuitive counter-part in equation (2) from (Faruqui, et al., 2014) we can consider that the *anonymity* pushing term, equation (3) from (Mrkšić, et al., 2016), is similar.

Probably the most important difference between the work of (Faruqui, et al., 2014), (Mrkšić, et al., 2016) and the currently analyzed work of (Lengerich, Maas, & Potts, 2017) if the fact that we can use function-based relationship modelling. The authors of *Functional Retrofitting* propose in their paper two different approaches of the relation modelling – a linear relationship presented in Figure 6 and a basic fully connected neural network with one hidden layer.

$$\Psi_{\mathcal{Q}}(\mathcal{Q};\mathcal{F}) = \sum_{i=1}^{n} \alpha_i \|\boldsymbol{q}_i - \hat{\boldsymbol{q}}_i\|^2 + \sum_{(i,j,r)\in\mathcal{E}} \beta_{i,j,r} \|\boldsymbol{A}_r\boldsymbol{q}_j + \boldsymbol{b}_r - \boldsymbol{q}_i\|^2 - \sum_{(i,j,r)\in\mathcal{E}^-} \beta_{i,j,r} \|\boldsymbol{A}_r\boldsymbol{q}_j + \boldsymbol{b}_r - \boldsymbol{q}_i\|^2 + \lambda \sum_{r\in\mathcal{R}} \|\boldsymbol{A}_r\|^2$$

Figure 6 - Linear relationship proposed by the authors of Functional Retrofitting. Applying *β=0* for *negative space* $\mathcal{E}^-$ and using identity as the value for the linear equation coefficient *A* gives us the objective function proposed by (Faruqui, et al., 2014)

Maybe one of the most important take-aways of this particular paper is exactly the cross-domain application – starting from semantic vector space models and knowledge graphs in general and finally using the *Functional Retrofitting* approach to identify potential uses for existing drugs in new diseases where those drugs were not yet employed. The authors make the source code available at `https://github.com/roaminsight/roamresearch`.

2.4 Fine-tuning representations generation

The last paper we will analyze in the current literature review for the proposed representation learning *product replacement pipeline* related work is the *Mittens* extension of the *GloVe* algorithm for semantic vector space models generation (Dingwall & Potts, 2018). The main intuition proposed by the authors of this work is that a pre-trained semantic vector space model can be further extended with new vector representations as well as update the existing ones with the new specialized domain data. The

authors start from the classic *GloVe* algorithm (Pennington, Socher, & Manning, 2014) and extend it in such a way that a new vector space model is generated from a previous *GloVe* set of embeddings, a new vocabulary and a new matrix of co-occurence. This method preserves the pre-trained *GloVe* vector space model relationships as well as adapts previous embeddings to new relationships and create new embeddings for previously unseen items. We can see a clear relationship between this approach and the previously presented *retrofitting* approaches, however this new framework choses to address directly both the problem of retrofitting and new embeddings generation rather than just fine-tuning existing embeddings.

The objective function of the proposed model consists of two parts: the first one being the exact *GloVe* objective function from equation (1) applied to the "new" co-occurrence matrix and the second part being the vector space preservation equation similar to the retrofitting cousins from equations (5) and (8).

During our pipeline research and development experiment we plan on applying both direct *retrofitting*-based fine-tuning on the pre-trained product embeddings as well as re-construction of the *GloVe* vector space with new co-occurrence matrices.

References

Dingwall, N., & Potts, C. (2018). *Mittens: An Extension of GloVe for Learning Domain-Specialized*. arXiv preprint arXiv:1803.09901.

Faruqui, M., Dodge, J., Jauhar, S. K., Dyer, C., Hovy, E., & Smith, N. A. (2014). *Retrofitting word vectors to semantic lexicons.* arXiv preprint arXiv:1411.4166.

Ganitkevitch, J., Van Durme, B., & Callison-Burch, C. (2013). PPDB: The paraphrase database. *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, (pp. 758-764).

Grbovic, M., Radosavljevic, V., Djuric, N., Bhamidipati, N., Savla, J., Bhagwan, V., & Sharp, D. (2015). E-commerce in your inbox: Product recommendations at scale. *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, 1809-1818.

Le, Q., & Mikolov, T. (2014). Distributed representations of sentences and documents. *International conference on machine learning*, (pp. 1188-1196).

Lengerich, B. J., Maas, A. L., & Potts, C. (2017). *Retrofitting distributional embeddings to knowledge graphs with functional relations.* arXiv preprint arXiv:1708.00112.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Mikolov, Tomas, et al. "Efficient estimation of word representations in vector space.* arXiv preprint arXiv:1301.3781 .

Miller, G. A. (1995). WordNet: a lexical database for English. *Communications of the ACM*, 39-41.

Mrkšić, N., Séaghdha, D. O., Thomson, B., Gašić, M., Rojas-Barahona, L., Su, P. H., & Young, S. (2016). *Counter-fitting word vectors to linguistic constraints.* arXiv preprint arXiv:1603.00892.

Pennington, J., Socher, R., & Manning, C. (2014). GloVe: Global Vectors forWord Representation. *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, (pp. 1532-1543).

Volkovs, M., Yu, G. W., & Poutanen, T. (2017). Content-based Neighbor Models for Cold Start. *In Proceedings of the Recommender Systems Challenge 2017*, (pp. 1-6).