A Non-negative Tensor Factorization Model for Selectional Preference Induction

Tim Van de Cruys
University of Groningen
The Netherlands
`t.van.de.cruys@rug.nl`

Abstract

Distributional similarity methods have proven to be a valuable tool for the induction of semantic similarity. Up till now, most algorithms use two-way co-occurrence data to compute the meaning of words. Co-occurrence frequencies, however, need not be pairwise. One can easily imagine situations where it is desirable to investigate co-occurrence frequencies of three modes and beyond. This paper will investigate a tensor factorization method called non-negative tensor factorization to build a model of three-way co-occurrences. The approach is applied to the problem of selectional preference induction, and automatically evaluated in a pseudo-disambiguation task. The results show that non-negative tensor factorization is a promising tool for NLP.

1 Introduction

Distributional similarity methods have proven to be a valuable tool for the induction of semantic similarity. The aggregate of a word's contexts generally provides enough information to compute its meaning, viz. its semantic similarity or relatedness to other words.

Up till now, most algorithms use two-way co-occurrence data to compute the meaning of words. A word's meaning might for example be computed by looking at:

- the various documents that the word appears in (words × documents);

- a bag of words context window around the word (words × context words);

- the dependency relations that the word appears with (words × dependency relations).

The extracted data – representing the co-occurrence frequencies of two different entities – is encoded in a matrix. Co-occurrence frequencies, however, need not be pairwise. One can easily imagine situations where it is desirable to investigate co-occurrence frequencies of three modes and beyond. In an information retrieval context, one such situation might be the investigation of *words × documents × authors*. In an NLP context, one might want to investigate *words × dependency relations × bag of word context words*, or *verbs × subjects × direct objects*.

Note that it is not possible to investigate the three-way co-occurrences in a matrix representation form. It is possible to capture the co-occurrence frequencies of a verb with its subjects and its direct objects, but one cannot capture the co-occurrence frequencies of the verb appearing with the subject and the direct object *at the same time*. When the actual three-way co-occurrence data is 'matricized', valuable information is thrown-away. To be able to capture the mutual dependencies among the three modes, we will make use of a generalized *tensor* representation.

Two-way co-occurrence models (such as latent semantic analysis) have often been augmented with some form of dimensionality reduction in order to counter noise and overcome data sparseness. We will also make use of a dimensionality reduction algorithm appropriate for tensor representations.

2 Previous Work

2.1 Selectional Preferences & Verb Clustering

Selectional preferences have been a popular research subject in the NLP community. One of the first to automatically induce selectional preferences from corpora was Resnik (1996). Resnik generalizes among nouns by using WordNet noun

synsets as clusters. He then calculates the *selectional preference strength* of a specific verb in a particular relation by computing the Kullback-Leibler divergence between the cluster distribution of the verb and the aggregate cluster distribution. The *selectional association* is then the contribution of the cluster to the verb's preference strength. The model's generalization relies entirely on WordNet; there is no generalization among the verbs.

The research in this paper is related to previous work on clustering. Pereira et al. (1993) use an information-theoretic based clustering approach, clustering nouns according to their distribution as direct objects among verbs. Their model is a one-sided clustering model: only the direct objects are clustered, there is no clustering among the verbs.

Rooth et al. (1999) use an EM-based clustering technique to induce a clustering based on the co-occurrence frequencies of verbs with their subjects and direct objects. As opposed to the method of Pereira et al. (1993), their model is two-sided: the verbs as well as the subjects/direct objects are clustered. We will use a similar model for evaluation purposes.

Recent approaches using distributional similarity methods for the induction of selectional preferences are the ones by Erk (2007), Bhagat et al. (2007) and Basili et al. (2007).

This research differs from the approaches mentioned above by its use of multi-way data: where the approaches above limit themselves to two-way co-occurrences, this research will focus on co-occurrences for multi-way data.

2.2 Factorization Algorithms

2.2.1 Two-way Factorizations

One of the best known factorization algorithms is principal component analysis (PCA, Pearson (1901)). PCA transforms the data into a new coordinate system, yielding the best possible fit in a least square sense given a limited number of dimensions. Singular value decomposition (SVD) is the generalization of the eigenvalue decomposition used in PCA (Wall et al., 2003).

In information retrieval, singular value decomposition has been applied in latent semantic analysis (LSA, Landauer and Dumais (1997), Landauer et al. (1998)). In LSA, a term-document matrix is created, containing the frequency of each word in a specific document. This matrix is then de-composed into three other matrices with SVD. The most important dimensions that come out of the SVD allegedly represent 'latent semantic dimensions', according to which nouns and documents can be represented more efficiently.

LSA has been criticized for a number of reasons, one of them being the fact that the factorization contains negative numbers. It is not clear what negativity on a semantic scale should designate. Subsequent methods such as probabilistic latent semantic analysis (PLSA, Hofmann (1999)) and non-negative matrix factorization (NMF, Lee and Seung (2000)) remedy these problems, and indeed get much more clear-cut semantic dimensions.

2.2.2 Three-way Factorizations

To be able to cope with three-way data, several algorithms have been developed as multilinear generalizations of the SVD. In statistics, three-way component analysis has been extensively investigated (for an overview, see Kiers and van Mechelen (2001)). The two most popular methods are parallel factor analysis (PARAFAC, Harshman (1970), Carroll and Chang (1970)) and three-mode principal component analysis (3MPCA, Tucker (1966)), also called higher order singular value decomposition (HOSVD, De Lathauwer et al. (2000)). Three-way factorizations have been applied in various domains, such as psychometry and image recognition (Vasilescu and Terzopoulos, 2002). In information retrieval, three-way factorizations have been applied to the problem of link analysis (Kolda and Bader, 2006).

One last important method dealing with multi-way data is non-negative tensor factorization (NTF, Shashua and Hazan (2005)). NTF is a generalization of non-negative matrix factorization, and can be considered an extension of the PARAFAC model with the constraint of non-negativity (cfr. infra).

One of the few papers that has investigated the application of tensor factorization for NLP is Turney (2007), in which a three-mode tensor is used to compute the semantic similarity of words. The method achieves 83.75% accuracy on the TOEFL synonym questions.

3 Methodology

3.1 Tensors

Distributional similarity methods usually represent co-occurrence data in the form of a *matrix*. This form is perfectly suited to represent two-way co-occurrence data, but for co-occurrence data beyond two modes, we need a more general representation. The generalization of a matrix is called a *tensor*. A tensor is able to encode co-occurrence data of any *n* modes. Figure 1 shows a graphical comparison of a matrix and a tensor with three modes – although a tensor can easily be generalized to more than three modes.
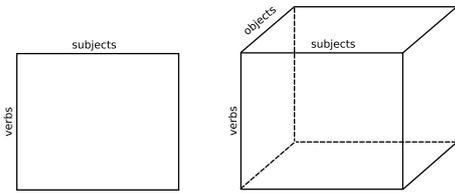

Figure 1: Matrix representation vs. tensor representation

3.2 Non-negative Tensor Factorization

In order to create a succinct and generalized model of the extracted data, a statistical dimensionality reduction technique called non-negative tensor factorization (NTF) is applied to the data. The NTF model is similar to the PARAFAC analysis – popular in areas such as psychology and bio-chemistry – with the constraint that all data needs to be non-negative (i.e. ≥ 0).

Parallel factor analysis (PARAFAC) is a multilinear analogue of the singular value decomposition (SVD) used in latent semantic analysis. The key idea is to minimize the sum of squares between the original tensor and the factorized model of the tensor. For the three mode case of a tensor $T \in \mathbb{R}^{D_1 \times D_2 \times D_3}$ this gives equation 1, where k is the number of dimensions in the factorized model and $\circ$ denotes the outer product.

$$\min_{x_i \in \mathbb{R}^{D1}, y_i \in \mathbb{R}^{D2}, z_i \in \mathbb{R}^{D3}} \| T - \sum_{i=1}^{k} x_i \circ y_i \circ z_i \|_F^2 \qquad (1)$$

With non-negative tensor factorization, the non-negativity constraint is enforced, yielding a model like the one in equation 2:

$$\min_{x_i \in \mathbb{R}^{D1}_{\geq 0}, y_i \in \mathbb{R}^{D2}_{\geq 0}, z_i \in \mathbb{R}^{D3}_{\geq 0}} \| T - \sum_{i=1}^{k} x_i \circ y_i \circ z_i \|_F^2 \qquad (2)$$

The algorithm results in three matrices, indicating the loadings of each mode on the factorized dimensions. The model is represented graphically in figure 2, visualizing the fact that the PARAFAC decomposition consists of the summation over the outer products of *n* (in this case three) vectors.
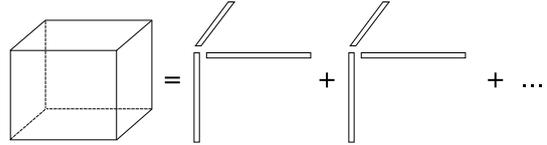

Figure 2: Graphical representation of the NTF as the sum of outer products

Computationally, the non-negative tensor factorization model is fitted by applying an alternating least-squares algorithm. In each iteration, two of the modes are fixed and the third one is fitted in a least squares sense. This process is repeated until convergence.[1]

3.3 Applied to Language Data

The model can straightforwardly be applied to language data. In this part, we describe the factorization of *verbs × subjects × direct objects* co-occurrences, but the example can easily be substituted with other co-occurrence information. Moreover, the model need not be restricted to 3 modes; it is very well possible to go to 4 modes and beyond — as long as the computations remain feasible.

The NTF decomposition for the *verbs × subjects × direct objects* co-occurrences into the three loadings matrices is represented graphically in figure 3. By applying the NTF model to three-way (s, v, o) co-occurrences, we want to extract a generalized selectional preference model, and eventually even induce some kind of frame semantics (in the broad sense of the word).

In the resulting factorization, each verb, subject and direct object gets a loading value for each factor dimension in the corresponding loadings matrix. The original value for a particular (s, v, o)

---

[1]The algorithm has been implemented in MATLAB, using the Tensor Toolbox for sparse tensor calculations (Bader and Kolda, 2007).
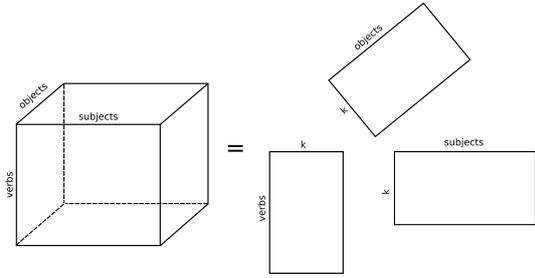
Figure 3: Graphical representation of the NTF for language data

triple x_{svo} can then be reconstructed with equation 3.

$$x_{svo} = \sum_{i=1}^{k} s_{si} v_{vi} o_{oi} \qquad (3)$$

To reconstruct the selectional preference value for the triple $(man, bite, dog)$, for example, we look up the subject vector for *man*, the verb vector for *bite* and the direct object vector for *dog*. Then, for each dimension i in the model, we multiply the ith value of the three vectors. The sum of these values is the final preference value.

4 Results

4.1 Setup

The approach described in the previous section has been applied to Dutch, using the Twente Nieuws Corpus (Ordelman, 2002), a 500M words corpus of Dutch newspaper texts. The corpus has been parsed with the Dutch dependency parser Alpino (van Noord, 2006), and three-way co-occurrences of verbs with their respective subject and direct object relations have been extracted. As dimension sizes, the 1K most frequent verbs were used, together with the 10K most frequent subjects and 10K most frequent direct objects, yielding a tensor of 1K $\times$ 10K $\times$ 10K. The resulting tensor is very sparse, with only 0.0002% of the values being non-zero.

The tensor has been adapted with a straightforward extension of pointwise mutual information (Church and Hanks, 1990) for three-way co-occurrences, following equation 4. Negative values are set to zero.[2]

---

[2]This is not just an ad hoc conversion to enforce non-negativity. Negative values indicate a smaller co-occurrence probability than the expected number of co-occurrences. Setting those values to zero proves beneficial for similarity calculations (see e.g. Bullinaria and Levy (2007)).

$$MI3(x,y,z) = \log \frac{p(x,y,z)}{p(x)p(y)p(z)} \qquad (4)$$

The resulting matrix has been factorized into k dimensions (varying between 50 and 300) with the NTF algorithm described in section 3.2.

4.2 Examples

Table 1, 2 and 3 show example dimensions that have been found by the algorithm with $k = 100$. Each example gives the top 10 subjects, verbs and direct objects for a particular dimension, together with the score for that particular dimension. Table 1 shows the induction of a 'police action' frame, with police authorities as subjects, police actions as verbs and patients of the police actions as direct objects.

In table 2, a legislation dimension is induced, with legislative bodies as subjects[3], legislative actions as verbs, and mostly law (proposals) as direct objects. Note that some direct objects (e.g. 'minister') also designate persons that can be the object of a legislative act.

Table 3, finally, is clearly an exhibition dimension, with verbs describing actions of display and trade that art institutions (subjects) can do with works of art (objects).

These are not the only sensible dimensions that have been found by the algorithm. A quick qualitative evaluation indicates that about 44 dimensions contain similar, framelike semantics. In another 43 dimensions, the semantics are less clearcut (single verbs account for one dimension, or different senses of a verb get mixed up). 13 dimensions are not so much based on semantic characteristics, but rather on syntax (e.g. fixed expressions and pronomina).

4.3 Evaluation

The results of the NTF model have been quantitatively evaluated in a pseudo-disambiguation task, similar to the one used by Rooth et al. (1999). It is used to evaluate the generalization capabilities of the algorithm. The task is to judge which subject (s or s') and direct object (o or o') is more likely for a particular verb v, where (s, v, o) is a combination drawn from the corpus, and s' and o' are a subject and direct object randomly drawn from the corpus. A triple is considered correct if the algorithm prefers both s and o over their counterparts

---

[3]Note that VVD, D66, PvdA and CDA are Dutch political parties.

subjects	su_s	verbs	v_s	objects	obj_s
politie 'police'	.99	*houd_aan* 'arrest'	.64	*verdachte* 'suspect'	.16
agent 'policeman'	.07	*arresteer* 'arrest'	.63	*man* 'man'	.16
autoriteit 'authority'	.05	*pak_op* 'run in'	.41	*betoger* 'demonstrator'	.14
Justitie 'Justice'	.05	*schiet_dood* 'shoot'	.08	*relschopper* 'rioter'	.13
recherche 'detective force'	.04	*verdenk* 'suspect'	.07	*raddraaiers* 'instigator'	.13
marechaussee 'military police'	.04	*tref_aan* 'find'	.06	*overvaller* 'raider'	.13
justitie 'justice'	.04	*achterhaal* 'overtake'	.05	*Roemeen* 'Romanian'	.13
arrestatieteam 'special squad'	.03	*verwijder* 'remove'	.05	*actievoerder* 'campaigner'	.13
leger 'army'	.03	*zoek* 'search'	.04	*hooligan* 'hooligan'	.13
douane 'customs'	.02	*spoor_op* 'track'	.03	*Algerijn* 'Algerian'	.13

Table 1: Top 10 subjects, verbs and direct objects for the 'police action' dimension

subjects	su_s	verbs	v_s	objects	obj_s
meerderheid 'majority'	.33	*steun* 'support'	.83	*motie* 'motion'	.63
VVD	.28	*dien_in* 'submit'	.44	*voorstel* 'proposal'	.53
D66	.25	*neem_aan* 'pass'	.23	*plan* 'plan'	.28
Kamermeerderheid 'Chamber majority'	.25	*wijs_af* 'reject'	.17	*wetsvoorstel* 'bill'	.19
fractie 'party'	.24	*verwerp* 'reject'	.14	*hem* 'him'	.18
PvdA	.23	*vind* 'think'	.08	*kabinet* 'cabinet'	.16
CDA	.23	*aanvaard* 'accepts'	.05	*minister* 'minister'	.16
Tweede Kamer 'Second Chamber'	.21	*behandel* 'treat'	.05	*beleid* 'policy'	.13
partij 'party'	.20	*doe* 'do'	.04	*kandidatuur* 'candidature'	.11
Kamer 'Chamber'	.20	*keur_goed* 'pass'	.03	*amendement* 'amendment'	.09

Table 2: Top 10 subjects, verbs and direct objects for the 'legislation' dimension

s' and o' (so the (s, v, o) triple – that appears in the test corpus – is preferred over the triples (s', v, o'), (s', v, o) and (s, v, o')). Table 4 shows three examples from the pseudo-disambiguation task.

s	v	o	s'	o'
jongere 'youngster'	*drink* 'drink'	*bier* 'beer'	*coalitie* 'coalition'	*aandeel* 'share'
werkgever 'employer'	*riskeer* 'risk'	*boete* 'fine'	*doel* 'goal'	*kopzorg* 'worry'
directeur 'manager'	*zwaai* 'sway'	*scepter* 'sceptre'	*informateur* 'informer'	*vodka* 'wodka'

Table 4: Three examples from the pseudo-disambiguation evaluation task's test set

Four different models have been evaluated. The first two models are tensor factorization models. The first model is the NTF model, as described in section 3.2. The second model is the original PARAFAC model, without the non-negativity constraints.

The other two models are matrix factorization models. The third model is the non-negative matrix factorization (NMF) model, and the fourth model is the singular value decomposition (SVD). For these models, a matrix has been constructed that contains the pairwise co-occurrence frequencies of verbs by subjects as well as direct objects. This gives a matrix of 1K verbs by 10K subjects + 10K direct objects (1K × 20K). The matrix has been adapted with pointwise mutual information.

The models have been evaluated with 10-fold cross-validation. The corpus contains 298,540 different (s, v, o) co-occurrences. Those have been randomly divided into 10 equal parts. So in each fold, 268,686 co-occurrences have been used for training, and 29,854 have been used for testing. The accuracy results of the evaluation are given in table 5.

The results clearly indicate that the NTF model outperforms all the other models. The model achieves the best result with 300 dimensions, but the differences between the different NTF models are not very large – all attaining scores around 90%.

subjects	su_s	verbs	v_s	objects	obj_s
tentoonstelling 'exhibition'	.50	*toon* 'display'	.72	*schilderij* 'painting'	.47
expositie 'exposition'	.49	*omvat* 'cover'	.63	*werk* 'work'	.46
galerie 'gallery'	.36	*bevat* 'contain'	.18	*tekening* 'drawing'	.36
collectie 'collection'	.29	*presenteer* 'present'	.17	*foto* 'picture'	.33
museum 'museum'	.27	*laat* 'let'	.07	*sculptuur* 'sculpture'	.25
oeuvre 'oeuvre'	.22	*koop* 'buy'	.07	*aquarel* 'aquarelle'	.20
Kunsthal	.19	*bezit* 'own'	.06	*object* 'object'	.19
kunstenaar 'artist'	.15	*zie* 'see'	.05	*beeld* 'statue'	.12
dat 'that'	.12	*koop_aan* 'acquire'	.05	*overzicht* 'overview'	.12
hij 'he'	.10	*in huis heb* 'own'	.04	*portret* 'portrait'	.11

Table 3: Top 10 subjects, verbs and direct objects for the 'exhibition' dimension

	dimensions		
	50 (%)	100 (%)	300 (%)
NTF	**89.52** ± 0.18	**90.43** ± 0.14	**90.89** ± 0.16
PARAFAC	85.57 ± 0.25	83.58 ± 0.59	80.12 ± 0.76
NMF	81.79 ± 0.15	78.83 ± 0.40	75.74 ± 0.63
SVD	69.60 ± 0.41	62.84 ± 1.30	45.22 ± 1.01

Table 5: Results of the 10-fold cross-validation for the NTF, PARAFAC, NMF and SVD model for 50, 100 and 300 dimensions (averages and standard deviation)

The PARAFAC results indicate the fitness of tensor factorization for the induction of three-way selectional preferences. Even without the constraint of non-negativity, the model outperforms the matrix factorization models, reaching a score of about 85%. The model deteriorates when more dimensions are used.

Both matrix factorization models perform worse than their tensor factorization counterparts. The NMF still scores reasonably well, indicating the positive effect of the non-negativity constraint. The simple SVD model performs worst, reaching a score of about 70% with 50 dimensions.

5 Conclusion and Future Work

This paper has presented a novel method that is able to investigate three-way co-occurrences. Other distributional methods deal almost exclusively with pairwise co-occurrences. The ability to keep track of multi-way co-occurrences opens up new possibilities and brings about interesting results. The method uses a factorization model – non-negative tensor factorization – that is suitable for three way data. The model is able to generalize among the data and overcome data sparseness.

The method has been applied to the problem of selectional preference induction. The results indicate that the algorithm is able to induce selectional preferences, leading to a broad kind of frame semantics. The quantitative evaluation shows that use of three-way data is clearly beneficial for the induction of three-way selectional preferences. The tensor models outperform the simple matrix models in the pseudo-disambiguation task. The results also indicate the positive effect of the non-negativity constraint: both models with non-negative constraints outperform their non-constrained counterparts.

The results as well as the evaluation indicate that the method presented here is a promising tool for the investigation of NLP topics, although more research and thorough evaluation are desirable.

There is quite some room for future work. First of all, we want to further investigate the usefulness of the method for selectional preference induction. This includes a deeper quantitative evaluation and a comparison to other methods for selectional preference induction. We also want to include other dependency relations in our model, apart from subjects and direct objects.

Secondly, there is room for improvement and further research with regard to the tensor factorization model. The model presented here minimizes the sum of squared distance. This is, however, not the only objective function possible. Another possibility is the minimization of the Kullback-Leibler divergence. Minimizing the sum of squared distance assumes normally distributed data, and language phenomena are rarely normally distributed. Other objective functions – such as the minimization of the Kullback-Leibler divergence – might be able to capture the language structures

much more adequately. We specifically want to stress this second line of future research as one of the most promising and exciting ones.

Finally, the model presented here is not only suitable for selectional preference induction. There are many problems in NLP that involve three-way co-occurrences. In future work, we want to apply the NTF model presented here to other problems in NLP, the most important one being word sense discrimination.

Acknowledgements

References

Brett W. Bader and Tamara G. Kolda. 2006. Efficient MATLAB computations with sparse and factored tensors. Technical Report SAND2006-7592, Sandia National Laboratories, Albuquerque, NM and Livermore, CA, December.

Brett W. Bader and Tamara G. Kolda. 2007. Matlab tensor toolbox version 2.2. `http://csmr.ca.sandia.gov/~tgkolda/TensorToolbox/`, January.

Roberto Basili, Diego De Cao, Paolo Marocco, and Marco Pennacchiotti. 2007. Learning selectional preferences for entailment or paraphrasing rules. In *Proceedings of RANLP 2007*, Borovets, Bulgaria.

Rahul Bhagat, Patrick Pantel, and Eduard Hovy. 2007. Ledir: An unsupervised algorithm for learning directionality of inference rules. In *Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP-07)*, pages 161–170, Prague, Czech Republic.

John A. Bullinaria and Joseph P. Levy. 2007. Extracting semantic representations from word co-occurrence statistics: A computational study. *Behavior Research Methods*, 39:510–526.

J. D. Carroll and J.-J. Chang. 1970. Analysis of individual differences in multidimensional scaling via an n-way generalization of "eckart-young" decomposition. *Psychometrika*, 35:283–319.

Kenneth Ward Church and Patrick Hanks. 1990. Word association norms, mutual information & lexicography. *Computational Linguistics*, 16(1):22–29.

Lieven De Lathauwer, Bart De Moor, and Joos Vandewalle. 2000. A multilinear singular value decomposition. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1253–1278.

Katrin Erk. 2007. A simple, similarity-based model for selectional preferences. In *Proceedings of ACL 2007*, Prague, Czech Republic.

R.A. Harshman. 1970. Foundations of the parafac procedure: models and conditions for an "explanatory" multi-mode factor analysis. In *UCLA Working Papers in Phonetics*, volume 16, pages 1–84, Los Angeles. University of California.

Thomas Hofmann. 1999. Probabilistic latent semantic analysis. In *Proc. of Uncertainty in Artificial Intelligence, UAI'99*, Stockholm.

H.A.L Kiers and I. van Mechelen. 2001. Three-way component analysis: Principles and illustrative application. *Psychological Methods*, 6:84–110.

Tamara Kolda and Brett Bader. 2006. The TOPHITS model for higher-order web link analysis. In *Workshop on Link Analysis, Counterterrorism and Security*.

Thomas Landauer and Susan Dumais. 1997. A solution to Plato's problem: The Latent Semantic Analysis theory of the acquisition, induction, and representation of knowledge. *Psychology Review*, 104:211–240.

Thomas Landauer, Peter Foltz, and Darrell Laham. 1998. An Introduction to Latent Semantic Analysis. *Discourse Processes*, 25:295–284.

Daniel D. Lee and H. Sebastian Seung. 2000. Algorithms for non-negative matrix factorization. In *NIPS*, pages 556–562.

R.J.F. Ordelman. 2002. Twente Nieuws Corpus (TwNC), August. Parlevink Language Technology Group. University of Twente.

K. Pearson. 1901. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2(6):559–572.

Fernando Pereira, Naftali Tishby, and Lillian Lee. 1993. Distributional clustering of English words. In *31st Annual Meeting of the ACL*, pages 183–190.

Philip Resnik. 1996. Selectional Constraints: An Information-Theoretic Model and its Computational Realization. *Cognition*, 61:127–159, November.

Mats Rooth, Stefan Riezler, Detlef Prescher, Glenn Carroll, and Franz Beil. 1999. Inducing a semantically annotated lexicon via em-based clustering. In *37th Annual Meeting of the ACL*.

Amnon Shashua and Tamir Hazan. 2005. Non-negative tensor factorization with applications to statistics and computer vision. In *ICML '05: Proceedings of the 22nd international conference on*

Machine learning, pages 792–799, New York, NY, USA. ACM.

L.R. Tucker. 1966. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31:279–311.

Peter D. Turney. 2007. Empirical evaluation of four tensor decomposition algorithms. Technical Report ERB-1152, National Research Council, Institute for Information Technology.

Gertjan van Noord. 2006. At Last Parsing Is Now Operational. In Piet Mertens, Cedrick Fairon, Anne Dister, and Patrick Watrin, editors, *TALN06. Verbum Ex Machina. Actes de la 13e conference sur le traitement automatique des langues naturelles*, pages 20–42, Leuven.

M. Alex O. Vasilescu and Demetri Terzopoulos. 2002. Multilinear analysis of image ensembles: Tensorfaces. In *ECCV*, pages 447–460.

Michael E. Wall, Andreas Rechtsteiner, and Luis M. Rocha, 2003. *Singular Value Decomposition and Principal Component Analysis*, chapter 5, pages 91–109. Kluwel, Norwell, MA, Mar.