Semi-supervised Data Clustering with Coupled Non-negative Matrix Factorization: Sub-category Discovery of Noun Phrases in NELL's Knowledge Base

Chunlei Liu
Machine Learning Department
Department of Physics
Carnegie Mellon University
chl56@andrew.cmu.edu

Tom Mitchell
Machine Learning Department
Carnegie Mellon University
tom.mitchell@cs.cmu.edu

Abstract

The standard non-negative matrix factorization (NMF) is a popular method to obtain low-rank approximation of a non-negative matrix, which is also powerful for clustering and classification in machine learning. In NMF each data sample is represented by a vector of features of the same dimension. In practice, we often have good side information for a subset of data samples. These side information might be binary vectors that indicate human provided class labels, or generic vectors in a different feature space. In this paper we propose the coupled non-negative matrix factorization (CNMF) method to automatically incorporate the side information of a subset of data. In CNMF, the matrix for data samples with or without side information in the original feature space and the matrix for data samples with side information in the new feature space are coupled together and iteratively optimized. Because of different qualities of the side information, a trade-off parameter is introduced to determine the importance of the side information, and we give a cross validation method to choose its value. The time complexity of the CNMF method could be several times bigger than the original NMF method, but still in the same order. As an example of implementing the CNMF method, we look into the knowledge base of the CMU Never-Ending Language Learning (NELL) project and find sub-categories of noun phrases.

1 Introduction

Application of clustering with multi-dimensional data is important in many fields, including information retrieval, text mining, image grouping and medical diagnosis. For many situations, we could define non-negative measurements of different features as data set $X = \{x_1, ..., x_n\} \in R_+^{m \times n}$, where each of the n samples is a m-dimensional non-negative feature vector. The standard clustering is usually unsupervised, where we hope to find the natural grouping of data samples based on measured or perceived similarities in the feature space. We explore methods to improve the clustering performance based on good side information on a subset of data samples, which is affordable in many cases. The side information could be class labels provided by domain experts, or more expensive and better measurements in a different feature space. As long as the side information is clean and allows easy clustering of the subset of data samples, the original clustering becomes semi-supervised and the performance can be improved. The side information can be represented by a matrix as $Y = \{y_1, ...y_{n'}\} \in R_+^{m' \times n'}$, where $n' < n$ and each data sample is a m'-dimensional non-negative feature vector. For example, if the side information are class labels, each data sample

in Y will a binary vector with the jth entry being 1 and the remaining elements being 0 if it is from class j.

As a case study, we cluster noun phrases from the knowledge base of the Never-Ending Language Learning (NELL) project at CMU [1]. Since it was launched in January 2010, NELL has extracted millions of noun phrases and relation instances by reading the web. There are currently more than one hundred top noun categories in NELL's knowledge base, such as `Species, Chemical, Religion, Economic Sector, and Emotion`. In the very beginning, the initial ontology of noun categories and relations are given. As times goes, NELL populates the noun categories and relations by learning new instances. For each main noun category, as there are more and more noun phrases learned, it is useful to discover good sub-categories with clustering methods, such that new noun phrases can be better learned and classified. For example, the `Animal` category has more than twenty thousand instances currently. In Tab. 1 we also show some noun phrase instances under the "movie" and "plant" category in NELL's knowledge base.

The input data set X for noun phrases from NELL is the co-occurrence matrix between noun phrase and context from input sentences. Both the noun phrase and context are defined using part-of-speech tag sequences. Each context can be viewed as a feature of the noun phrases, and the co-occurrence statistics is the measurement. The initial size of the context feature space is enormous, so in order to decrease the dimension and reduce computational time later, we only keep contexts that have co-occurrence bigger than a defined threshold. Further feature selections can also be applied, such as choosing the top ranked context features according to the co-occurrence with the noun phrases. Side information on part of the noun phrases can be obtained using relation instances between noun categories from NELL. For example, there are about two thousand instances in the relation `(arg1) animal_is_type_of_animal (arg2)`, where both arguments are from the animal category. Instances in this relation category, e.g., `(robin) is type of (bird)`, have subset/superset relations, as the first argument is a member of the second argument. Representing all the first arguments as columns and all the second arguments as features in rows, we can build the side information matrix Y, where each cell is either zero or one.

Category	Noun phrase instances
Movie	`aliens; an_american_in_paris; beowulf; big_lebowski; boogie_nights; close_encounters; collateral; dick_tracy; die_hard; doctor_zhivago; empire_strikes_back; ferris_bueller__s_day_off; fight_club; forrest_gump; goblet_of_fire; godzilla; goodfellas; good_will_hunting; hamlet hook; .`
Plant	`spruces; alders; algae; allium; annuals; apple_tree; apple_trees; aquatic_plants; aquatic_weeds; ash; ash_trees; aspen; aspens; asters; azalea; azaleas; bamboo; banana_trees; basswood; bean; .`

Table 1: Examples of categories and noun phrases in NELL's knowledge base.

In the following sections, we first describe several traditional clustering methods. We then focus on the clustering method based on NMF and how to improve it with CNMF, and we demonstrate the performance of CNMF through studies of synthesized data. In the following experiment section, we first implement several feature selection methods, then show the results of using traditional clustering methods, NMF and CNMF, separately.

2 Traditional Clustering Methods

General clustering methods can be divided into two main categories: hierarchical and partitional. In a hierarchical algorithm, a dendrogram is created to represent the similarity levels at which groupings change. The most popular hierarchical algorithms include the single-link [2] and complete-link [3] algorithms. For the partitional clustering algorithm, on the contrary, a single partition of the data instead of a clustering structure is obtained. Since hierarchical algorithms are usually com-

putationally prohibitive, partitional methods have the advantages in application that involve large data.

The most frequently used criterion function in partitional clustering is the squared error criterion. This criterion tends to work well if the data is isolated and compact. The K-means algorithm [4] is the simplest algorithm that uses this squared error metric. We can also employ other distance metrics such as the cosine distance for the K-means, which might work better for certain data. Another commonly used partitional methods is the mixture model based algorithms. The underlying assumption is that the data to be clustered are drawn from several different distributions corresponding to different clusters. The goal of the algorithm is to identify the parameters for each distribution. To cluster data that has no real labels known, the Expectation Maximization (EM) algorithms can be applied [5].

2.1 K-means Algorithm

Given a set of observations $X = \{x_1, ..., x_n\} \in R_+^{m \times n}$, where each observation is a m-dimensional vector, K-means clustering algorithm partitions the n observations into K sets $S = \{S_1, S_2, ..., S_K\}$, such that the within-cluster sum of squared distances to the centroid is minimized:

$$\arg\min_S \sum_{i=1}^{K} \sum_{X_j \in S_i} ||X_j - \mu_i||^2, \tag{1}$$

where we have used the Euclidean distance, and μ_i is the mean of points in cluster S_i. To find the optimal cluster assignments, the K-means algorithm uses an iterative refinement technique. First, the data points are given some initial cluster assignments (e.g., random initialization). Then the algorithm proceeds by alternating between the following two steps until assignments of clusters no longer change:

- Assignment step: assign each data point to the cluster whose mean is closest to it;
- Update step: re-calculate the mean of each cluster as the centroid of that cluster.

The original K-means algorithm described here is known to find local optima only. To overcome this problem, we actually use the K-means++ algorithm which aims at a better initialization [6] to avoid being stuck at a local optima. The specific algorithm is as follows:

- The first cluster center is chosen uniformly at random from among all the data points.
- For each data point x, compute the distance $D(x)$ between x and the nearest center that has been chosen.
- Choose one new data point as a new center according to the probability that is proportional to $D^2(x)$.
- Repeat the second and third steps until all the needed centers have been chosen.
- With all the initial cluster centers chosen we proceed using the standard K-means clustering.

2.2 Multinomial Mixture Model

We use the multinomial mixture model with naive Bayes assumption to model the data we have. In the frame work of supervised learning, where the true cluster label of each instance is known, it is easy to estimate the parameters. A mixture model is a linear combination of models as:

$$P(X) = \sum_Z P(Z)P(X|Z)$$

where Z identifies the mixture component. $P(Z)$ is the probability of generating component Z, and $P(X|Z)$ is the distribution associated with the mixture component Z. In this project, we assume $P(X|Z)$ is a multinomial distribution, and $Z = 1, .., L$ are the cluster labels. After we learn $P(Z)$ and $P(X|Z)$, we can compute the cluster probabilities for each data item X_i as :

$$P(Z = Z_i|X = X_i) \propto P(Z = Z_i)P(X = X_i|Z = Z_i).$$

We then assign data X_i the label as $arg_{max}P(Z = Z_i|X = X_i)$. In order to apply the mixture model, we need to learn the parameters associated with probability density functions first. For $X = \{x_1, ..., x_n\} \in R_+^{m \times n}$ the row index is $j = \{1, .., m\}$ for each data sample X_i. The number of times row j co-occurs with X_i is $N_j(X_i)$. The parameters and probability density functions are linked through:

$$P(Z_i = k) = \phi_k$$
$$P(X_{i,j}|Z_i = k) = \theta_{k,j}$$
$$P(X_i, Z_i = k) = \phi_k \Pi_{k=1}^{L} \theta_{k,j}^{N_j(X_i)}$$

The sufficient statistics for estimating parameters in the multinomial mixture using the maximum likelihood method is as follows:

- $n_k = \sum_{i=1}^{n} I(Z_i, k)$, i.e., number of times clusters k is seen.
- $n_{k,j} = \sum_{i=1}^{n} N_j(X_i) I(Z_i, k)$, i.e., number of times context j is seen in cluster k.

If the cluster label of each X_i is known, the maximum likelihood gives estimates as:

$$\phi_k = \frac{n_k}{n} \tag{2}$$
$$\theta_{k,j} = \frac{n_{k,j}}{\sum_{j'=1}^{n} n_{k,j'}}$$

In our case, since we do not know the true label of each animal instance, we cannot estimate the parameters with the training data. However, we can still obtain estimations using the EM algorithms as the following:

- Guess the initial values $\phi^{(0)}$ and $\theta^{(0)}$
- Repeat the following, for iterations t=1,2,...:
  - E-step: calculate the expected values of sufficient statistics:

$$E[n_k] = \sum_{i=1}^{n} P_{\phi^{(t-1)}, \theta^{(t-1)}}(Z_i = k|X_i) \tag{3}$$
$$E[n_{k,j}] = \sum_{i=1}^{n} N_j(X_i) P_{\phi^{(t-1)}, \theta^{(t-1)}}(Z_i = k|X_i)$$

  - M-step: update the parameters based on sufficient statistics

$$\phi_k^{(t)} = \frac{E[n_k]}{n} \tag{4}$$
$$\theta_{k,j}^{(t)} = \frac{E[n_{k,j}]}{\sum_{j'=1}^{n} E[n_{k,j'}]}$$

- after convergence of previous step, assign label to each data X_i as $arg_{max}P(Z = Z_i|X = X_i)$.

2.3 Clustering Through Dimension Reduction

Clustering algorithms that are based on distance measure are often not effective for data with high dimensions, as distance between nearest points in high dimension is not different from that of other points [7]. On the other hand, in practice, high dimensional data usually have much lower intrinsic dimension which makes clustering possible and reduces computational cost. Methods such as the Principle component analysis (PCA) construct a linear combination of low dimensional vectors that can best describe the variance of the original data [8]. There are also non-linear methods such as the Multidimensional Scaling (MDS) [9] and the Spectral Clustering algorithm [10]. In the MDS method, the original high dimensional data are projected into a low dimensional structure while maintaining the proximity information. For the spectral clustering algorithm, the similarity matrix between different points in the data are constructed, and the clustering algorithm partitions the data into different subsets by optimizing certain criterion functions.

3 Clustering Method with Matrix Factorization and Related Work

Non-negative matrix factorization (NMF) is similar to the spectral clustering algorithm, and has been shown to be very useful in presenting non-negative data with intuitive basis vectors [11]. With the NMF method, we can obtain a lower rank-k approximation by factorizing the data matrix $X \in R_+^{m\times n}$ into two non-negative factor matrix $W \in R_+^{m\times k}$ and $H \in R_+^{k\times n}$ by minimizing the following cost function:

$$\min_{W\geq 0, H\geq 0} f(W,H) = \frac{1}{2}||X - WH||_F^2, \tag{5}$$

where F stands for Frobenius norm, W is referred as the basis matrix and H is called as encoding matrix. Given the rank k, the NMF aims at finding the best basis axis to project the original data onto a k-dimensional space. Columns of the basis matrix W gives the basis vectors, and columns of the encoding matrix H gives the membership of each new dimension for the data points. The optimization problem shown in Eqn. 5 is not convex, so it is hard to find the global minimum. However if W satisfies the separability condition, algorithms for exact solution with polynomial running time have been given [12]. In many cases finding local minimum is still useful. Various algorithms have been invented to find factor matrix W and H, including multiplicative update rule [13] and several algorithms based on alternating non-negative least squares, such as the projected gradient descent methods [14], the active-set method [15] and the block principal pivoting method [16].

The NMF method has been successfully used in clustering problems as in [17]. In the NMF space, each basis axis can be related to a cluster. Clustering the data points is equivalent as finding the axis with which the data point has the largest projection value in the encoding matrix H. We can also view the NMF as a dimension reduction method and further apply traditional clustering methods such as K-means on the column vectors in H.

Several semi-supervised NMF methods have been developed to incorporate prior knowledge and to improve the clustering performance [18],[19],[20]. Our proposed coupled non-negative matrix factorization (CNMF) method is distinguished from these existing work in several ways:

- While some methods use pairwise must-link or cannot-link constraints [18],[19], and other methods exploit class labels on a few data samples [20], our CNMF method assumes side information in a more general feature space. The pairwise constraints or class labels can be treated as a special case of feature vectors in CNMF. Notice that these constraints are soft constraints in CNMF, and the strength can be controlled by a trade-off parameter.
- Compared to method used in [20], no extra weight matrix is needed in CNMF to distinguish data samples which have side information available or unavailable. Instead, CNMF employs a sum of three residuals to mathematically and intuitively couple the data matrix without side information, the data matrix with side information and the side information matrix together.
- We develop updates for the projected gradient method as in [14], which are different from algorithms used in other methods [18],[19],[20] such as multiplicative updated.
- Similar to the method in [20], we introduce a trade-off parameter to determine the influence of the side information matrix, but we also introduce a cross validation method to choose the size of the parameter.

4 Clustering Method with Coupled Non-negative Matrix Factorization

First we notice that the side information matrix Y can be separately factorized as the data matrix X in Eqn. 5. In this case, we optimize the following objective function:

$$\min_{W_Y\geq 0, H_Y\geq 0} f(W_Y,H_Y) = \frac{1}{2}||Y - W_Y H_Y||_F^2. \tag{6}$$

Since X and Y share a common subset of data samples, we divide X into two parts X_1 and X_2, where X_1 and X_2 has the same number of rows (features) but different number of columns (data points). Columns of X_2 represent the same subset of data samples as in Y, while columns of X_1 represent all the remaining data samples. Thus factorizing X is equivalently as optimizing:

$$\min_{W_X\geq 0, H_1\geq 0, H_2\geq 0} f(W_X,H_1,H_2) = \frac{1}{2}||X_1 - W_X H_1||_F^2 + \frac{1}{2}||X_2 - W_X H_2||_F^2, \tag{7}$$

where we factor X_1 and X_2 onto the same basis matrix W_X but different encoding matrices H_1 and H_2. We further notice that matrix X_2 and Y represent the same subset of data samples in different feature spaces, but they are related by having the same encoding matrix with $H_Y = H_2$. Thus we obtain the final coupled optimization objective function as:

$$\min_{W_X \geq 0, W_Y \geq 0, H_1 \geq 0, H_2 \geq 0} f(W_X, W_Y, H_1, H_2) \tag{8}$$
$$= \frac{1}{2}||X_1 - W_X H_1||_F^2 + \frac{1}{2}||X_2 - W_X H_2||_F^2 + \frac{1}{2}||Y - W_Y H_2||_F^2,$$

where all the factor matrices are coupled in the same function such that good side information leads to good encoding matrix H_2 for data samples with side information, good H_2 matrix then helps to find good basis matrix W_X, and eventually good W_X results in good encoding matrix H_1 for the remaining data samples. After the solutions are obtained, we concatenate on H_1 and H_2 to get the encoding matrix for all data samples.

When optimizing the objective function in Eqn. 8, in order to balance different terms, it is better to first normalize X_1, X_2 and Y by column with some normalization method such as L2-norm. Considering different qualities of the side information in Y, we can also introduce a trade-off parameter λ:

$$\min_{W_X \geq 0, W_Y \geq 0, H_1 \geq 0, H_2 \geq 0} f(W_X, W_Y, H_1, H_2) \tag{9}$$
$$= \frac{1}{2}||X_1 - W_X H_1||_F^2 + \frac{1}{2}||X_2 - W_X H_2||_F^2 + \frac{\lambda}{2}||Y - W_Y H_2||_F^2,$$

where if $\lambda = 0$, the CNMF is same as the original NMF of X without any side information, while if λ gets bigger, the role of side information becomes more important in the CNMF.

Like the original NMF, the CNMF problem shown in Eqn. 9 is a non-convex optimization problem. Here we apply the projected gradient algorithm based on ANLS as shown in Alg. 1 to find a local minimum. In order to solve the subproblems as in Eqn. $11-14$, we need to calculate the gradient and Hessian matrix for each factor matrix. One difference from the method in [9] is that we have H_2 coupled in two terms, where the gradient for H_2 is:

$$\nabla f(H_2) = (W_X^T W_X) H_2 - W_X^T X_2 + \lambda((W_Y^T W_Y) H_2 - W_Y^T Y),$$

and the Hessian matrix is block diagonal with each block equals $W_X^T W_X + \lambda W_Y^T W_Y$. To check the convergence of the algorithm, we compare the function value at each iteration to the value from the previous iteration. The convergence is achieved if the change is smaller than a defined threshold (e.g., 10^{-6}). The time complexity of this algorithm is similar to the original projected gradient algorithm. For each outer iteration, the original NMF with projected gradient algorithm need $O(tmk^2)$ to minimize H and $O(tnk^2)$ to minimize W, where t is the number of inner iterations. In CNMF, there are more terms to minimize for each outer iteration, so the cost could be several times bigger than the NMF without the side information.

The size of parameter λ in Eqn. 9 can be chosen with cross validation methods using some hold-out data. After the basis matrix W_X is obtained by factorizing the matrix X, we use it to factorize the hold-out data by optimizing the following objective function:

$$\min_{H_{hold} \geq 0} \frac{1}{2}||X_{hold} - W_X H_{hold}||_F^2, \tag{10}$$

where X_{hold} is the matrix for the hold-out data, and H_{hold} gives us the memberships of the hold-out data on different basis axis. The optimized function value of the hold-out data can be used to compare performances of factorizing X with different size of λ, where smaller function value means better factorization.

5 Studies with Synthesized Data

To simulate the data matrix X, we generate 600 data samples with 2000 features from 10 clusters using the following procedure. Every consecutive 60 samples are generated from the same cluster. For each cluster, we randomly pick 300 out of 2000 features. Then for each sample in the same

Algorithm 1 Alternating nonnegative least squares for CNMF

Initialize: $W_X^1 \geq 0, W_Y^1 \geq 0, H_1^1 \geq 0, H_2^1 \geq 0$
repeat
  for $k = 1$ **to** $1, 2, ...$ **do**

$$W_Y^{k+1} = \mathrm{a}rgmin_{W_Y \geq 0} f(W_X^k, H_1^k, W_Y, H_2^k) \quad (11)$$
$$H_1^{k+1} = \mathrm{a}rgmin_{H_1 \geq 0} f(W_X^k, H_1, W_Y^{k+1}, H_2^k) \quad (12)$$
$$H_2^{k+1} = \mathrm{a}rgmin_{H_2 \geq 0} f(W_X^k, H_1^{k+1}, W_Y^{k+1}, H_2) \quad (13)$$
$$W_X^{k+1} = \mathrm{a}rgmin_{W_X \geq 0} f(W_X, H_1^{k+1}, W_Y^{k+1}, H_2^{k+1}) \quad (14)$$

  end for
until convergence

cluster we randomly pick half of the 300 cluster features and assign them random values between 1 and 50, and assign 0 to all the remaining features. After the data are generated, we add a uniformly distributed random number between 1 and 100 to all feature values of all the data as noise. We simulate the side information matrix in three different ways: (1) We randomly choose 6 data samples from each cluster and define a new feature space of dimension 10. For all the 6 data samples coming from the same cluster, we set the jth feature value to be 1 and 0 for the remaining features if the data sample is from the jth cluster. This corresponds to giving the class labels to part of the data samples. We denote this side information matrix as Y_1. (2) We randomly choose 6 data samples from each cluster and define a new feature space of dimension 300. For all the 6 data samples coming from the same cluster, we randomly select 30 features and set their value to be 1 and 0 for all the remaining features. This corresponds to good measurements of part of the data samples in a new feature space. We denote this side information matrix as Y_2. (3) We randomly choose 6 data samples from each cluster and define a new feature space with dimension 300. For each data sample, we set all of their feature values to be random numbers between 0 and 1. This corresponds to noisy and completely uninformative side information. We denote this side information matrix as Y_3.

In order to find a reasonable size of λ in Eqn. 9, we take 100 data samples as the hold-out data by choosing 10 samples from each cluster. Data samples which have side information are excluded when generating the hold-out data. The remaining 500 data samples are put into matrix X and we also have side information on 60 data samples. For each type of side information encoded in Y_1, Y_2 and Y_3, we vary the size of λ and apply algorithm 1 to optimize the objective function in Eqn. 9. With the factor matrices obtained, we then optimize the objective function in Eqn. 10 using the hold-out data set. The function value of the hold-out data set versus λ value are shown on the left of Fig. 1. According to the results, using good side information in Y_1 or Y_2 leads to better factorization of the hold-out data set as the value of λ increases. On the other hand, if the side information is random noise and thus useless, the factorization could get worse as we increase the value of λ. The benefits of using good side information in the CNMF can also be seen by visualizing the encoding matrix H. In Fig. 2, we show the transpose of the H matrix with Y_2 for the first six λ values. As the value of λ increase from 10^{-4} to 0.1, the 10 clusters can be seen more and more clearly.

In previous situations, we have side information for data samples from all the 10 clusters. In practice, we might only have side information for data samples from part of the clusters. We check the performance of the CNMF by simulating matrix Y from only 5 clusters. Specifically, we set up Y using the second method as previously discussed, but with side information for only 30 data samples from 5 clusters only (6 data samples from each cluster). We also take 100 randomly chosen data samples from each cluster as the hold-out data set, and use the remaining 500 data samples and side information on 30 data samples to implement the CNMF. With the results, we optimize the objective function in Eqn. 10 using the hold-out data set. The function value versus different size of λ is shown on the right of Fig. 1, where we can see that the factorization also gets better as the value of λ increases.

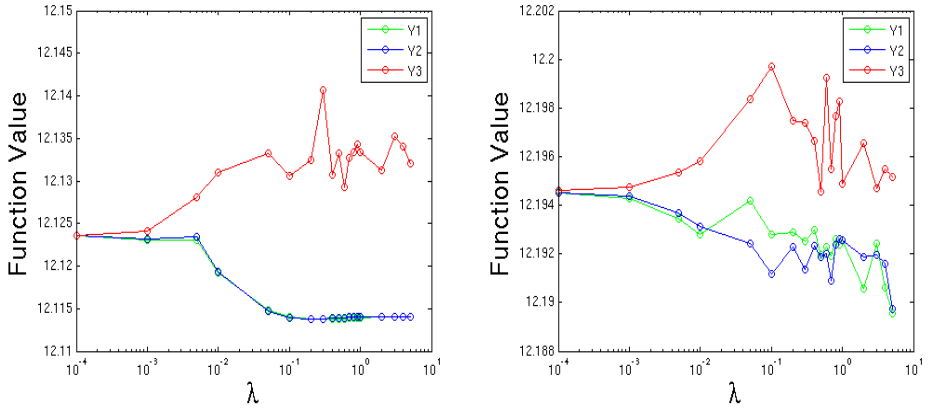

Figure 1: (left) Value of the objective function from Eqn. 10 versus λ with side information from all the 10 clusters, which is calculated on a hold out data set independent of the training set. (right) Value of the objective function as in Eqn. 10 versus λ using a hold out data set with side information from half of the 10 clusters .

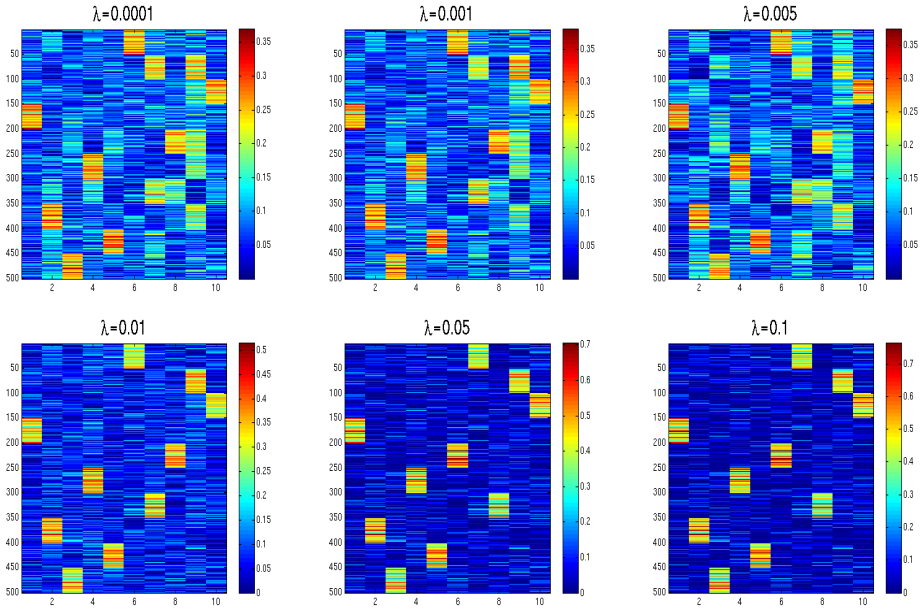

Figure 2: H^T (encoding matrix) of 500 training samples for different λ values, with good side information from Y_2.

6 Experiments

We look into NELL's knowledge base and use the co-occurrence matrix between noun phrase and context as the input data. As an example, we obtain the co-occurrence matrix for the `Animal` category which has about 20 thousand animal instances and about 5 million context features. We further choose the top 600 popular animal instances to serve as our training data set, where the popularity is ranked according to their total co-occurrences with all the context features.

In the following sections, we first describe different ways of selecting features. We then apply the Kmeans and EM algorithms to cluster the training data set. In order to obtain side information and

improve the cluster results, we look into the relation instances involving animal instances and apply the CNMF method.

6.1 Feature selection

Contexts of the co-occurrence matrix can be viewed as features of the animal instances. For the 600 training data, the initial dimension of the feature space is about 379 thousands. It is computationally expensive to perform the following clustering analysis, and the result might be degraded as well. In order to improve the clustering result and reduce computational burden, we apply different feature selection methods.

The first feature selection method is to rank contexts according to their total co-occurrences with all the animal instances. We then choose the contexts that have high ranks in the list as the features. As an example, we show the top 20 and bottom 20 contexts in Tab. 2.

Rank	Contexts
top 20	number of _; species of _; you have _; _ 's life; _ 's books; group of _; needs of _; care of _; thousands of _; I was _; _ is in; _ are in; variety of _; _ living in; _ 's name; kind of _; types of _; lot of _; parents of _; lives of _ .
bottom 20	likely consume _; _ has product development; _ 's lead drug; Iowa ; started _; care have made _; Fables From _; Unnatural History of _; manager , has been with _; _ 's Pizzeria; salt deposits at _; deposits at _; Alumni Association Distinguished _; Association ???s Distinguished _; _ moving hundreds; paste recipe for _; _ awarded since; Other companies using _; sales of western _; jet thompson _; Council features _.

Table 2: Examples of contexts ranked according to their total co-occurrence with all the animals.

As we can see from the results, it is easy to see that highly ranked contexts co-occur frequently with animal instances, while lowly ranked contexts does not have any meaningful relationship with animal instances. We further check some statistics associated with the resulting co-occurrence matrix. Taking the top 2,000 contexts as an example, the co-occurrence matrix has a dimension of 2000 by 600. In Fig. 3, we show the number of contexts out of the 2000 contexts each animal co-occur with, and the average number of co-occurrence for each animal with these contexts. On average, each animal instance co-occurs with 704 contexts with a mean co-occurrence frequency of 17.

In the first method, although highly ranked contexts based on total co-occurrence with all the animal instances are good for the whole category, they might be not very powerful to distinguish different sub-categories. An an improvement, we first choose a much larger subset of contexts with the first method, then we apply a different method to refine it. Here we propose the second method following [21] that defines the variance quality for the jth context as the following:

$$q(j) = \frac{1}{m}\sum_{i=1}^{m} f_{ij}^2 - \frac{1}{m^2}\left[\sum_{i=1}^{m} f_{ij}^2\right]^2, \tag{15}$$

where m is the total number of animal instances (600 in this case), and f_{ij} is the co-occurrence of between animal instance i and context j. The variance quality tells us the variances of co-occurrence between each context and all the animals. Intuitively, the larger the value of the variance is, the stronger distinction power the context has. We choose the top 100 thousands contexts using the first method, then we calculate the variance quality for all these 100 thousands contexts and rank them according to the variance quality. Here we list the top 20 and bottom 20 contexts in Tab. 3.

We further check the resulting co-occurrence matrix. Taking the top 2,000 contexts as an example, where the dimension of the co-occurrence matrix is 2000 by 600. In Fig. 4, we show the number

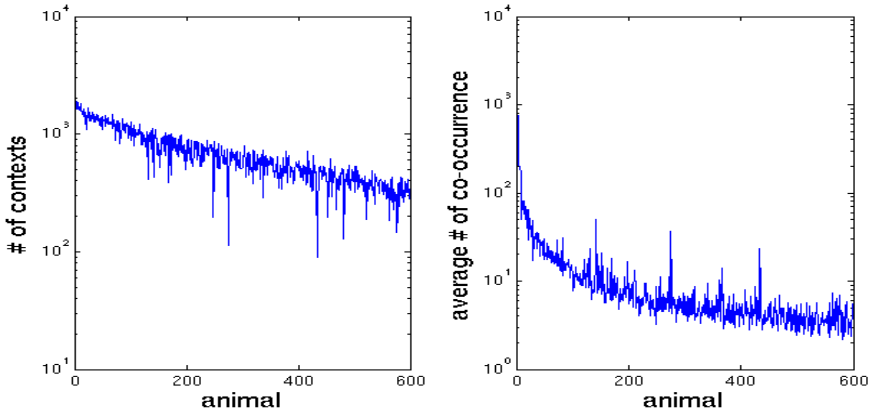

Figure 3: Ranking contexts according to the total co-occurrences with all animal: (left) Number of contexts out of 2000 top ranked contexts each animal instance co-occurs with, (right) average number of co-occurrence per context for each animal. In both plots, the animal instances are sorted according to the total co-occurrences with all the contexts.

Rank	Contexts
top 20	number of _; _ 's books; I was _; needs of _; _ 's Hospital; _ 's life; families with _; parents of _; Son of _; click of _; you have _; lives of _; reach of _; _ 's book; _ 's education; care of _; _ 's literature; _ 's share; _ 's hand; _ 's name.
bottom 20	_ are well known in; _ were moving into; same league as _; _ are as large; creature resembling _; _ still survive in; it did n't take long for _; _ filling the air; It was filled with _; Today , with _; complete lack of _; forest are _; _ smashed into; He is into _; road at _; juveniles of _; they are at _; successful breeding of _; Other names for _; group include _.

Table 3: Examples of contexts ranked according to their variance quality.

of contexts each animal co-occur with, and the average number of co-occurrence for each animal with all the contexts. In an overall average, each animal co-occur with 521 contexts with a mean co-occurrence frequency of 18.

We look into the third method for feature selection. In the NELL project, certain extraction patterns are used to identify new instances for each category. These extraction patterns are just a small subset of all the contexts, and they are learned and improved continuously as the NELL project progresses. Specifically [22], if a promoted category instance is found in a sentence, NELL extracts the preceding words as a candidate pattern if they are verbs followed by a sequence of adjectives, prepositions, determiners (e.g., "being acquired by arg1") or nouns and adjectives followed by a sequence of adjectives, prepositions, or determiners (e.g., "former CEO of arg1"). NELL also extracts the words following the instance as a candidate pattern if they are verbs followed optionally by a noun phrase (e.g., "arg1 broke the home run record"), or verbs followed by a preposition (e.g., "arg1 said that"). These extraction patterns should also serve as good features to identify instances within the same main category. As of December 2012, there are 1988 such extraction patterns for the animal category in the NELL knowledge base. With 600 animal instances in our test data, we thus obtain the co-occurrence matrix with dimension of 1988 by 600. In Fig. 5, we show the number of extraction patterns each animal co-occur with, and the average number of co-occurrence for each animal with

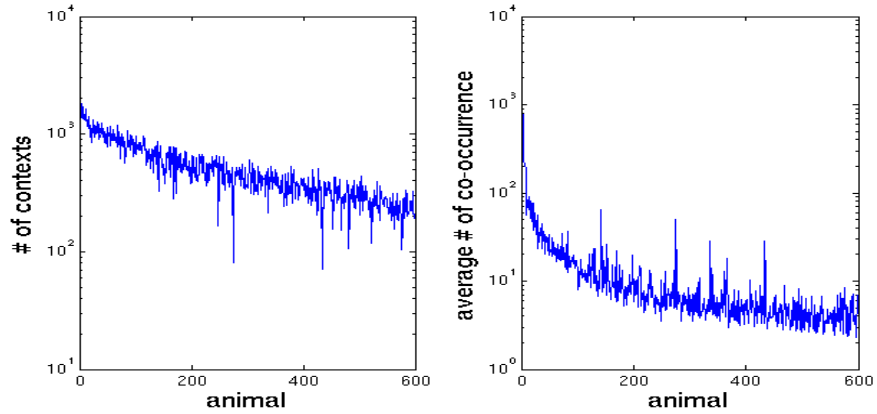

Figure 4: Ranking contexts according to co-occurrence variance: (left) Number of contexts out of 2000 top ranked contexts each animal instance co-occurs with, (right) average number of co-occurrence per context for each animal. In both plots, the animal instances are sorted according to the total co-occurrences with all the contexts.

all the extraction patterns. In an overall average, each animal co-occur with 51 extraction patterns with a mean co-occurrence frequency of 10. One thing to notice is that these average numbers are much smaller compared to those from the first two methods.

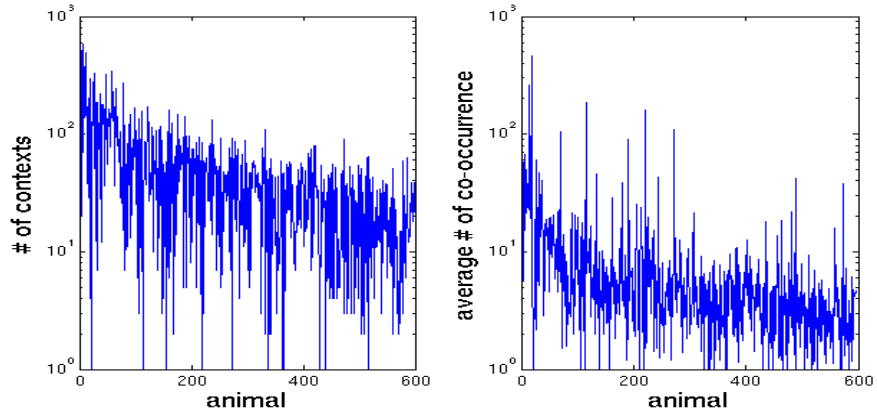

Figure 5: Using contexts that are extraction patterns for the animal category learned by NELL: (left) Number of contexts out of about 2000 extraction patterns each animal instance co-occurs with, (right) average number of co-occurrence per context for each animal. In both plots, the animal instances are sorted according to the total co-occurrences with all the contexts.

6.2 Clustering results from K-means++ algorithm

We first apply the K-means algorithm on the training data set. It is important to normalize the data first. Because animal instances that are more common tend to have a much larger co-occurrence with contexts, the relative co-occurrence matters more when we try to cluster them. Any reasonable normalization methods seem to work, and using the cosine distance instead of Euclidean distance is as good as the normalization method. Here we normalize the matrix such that each column is divided by its L2-norm. Using the variance quality as the feature selection method, we choose the top 2,000 contexts. We set the number of clusters to be 30 and apply the K-means++ algorithm. Many of the cluster results look reasonable, and some of the reasonable clusters found are shown in Tab. 4.

Class	Animal instance
7	cattle; cows; buffalo; elephant; elephants; goats; elk; camels; bison; reindeer; caribou; zebra; alpacas; llamas; dairy cows; beef cattle; dairy cattle; zebras; wildebeest; impala; mammoths; buffaloes; gazelle; yak; water buffalo .
14	salmon; bass; trout; cod; catfish; carp; walleye; perch; marlin; rainbow trout; snapper; steelhead; crappie; sturgeon; tarpon; halibut; stripers; chinook; flounder; atlantic salmon; redfish; snook; sailfish; lake trout; walleyes; brook trout; bonefish; bream; dorado; king salmon; coho; chinook salmon;.

Table 4: Examples of clustering results from the K-means++ algorithm, where within each cluster animal instances are randomly listed.

6.3 Clustering results from EM algorithm

Same as the K-means++ algorithm, we do not know the true number of clusters, but we can apply the Bayesian information criterion (BIC) to choose the appropriate number. The formula for BIC is defined as:

$$2\ln p(x|k) \approx BIC = 2\mathrm{L} - k\ln(n), \tag{16}$$

where L is the data likelihood, k is the number of free parameters, and n is the number of data points. In our case, $n = 600$ and $k = (m-1) * K + (K-1)$, where m is the number of context features (e.g. 2,000) and K is the proposed number of clusters. Using the variance quality as a feature selection method, we choose the top 2,000 contexts and apply the EM algorithm. The BIC versus different number of clusters (up to 100) is shown on the left of Fig. 6. According to BIC scores for different number of clusters, we can choose 30 as the optimal value. Fixing the number of clusters to be 30, we obtain the cluster results. The cluster size for each cluster is shown on the right of Fig. 6. Some of the reasonable clusters found are also shown in Tab. 5.

Class	Animal instance
12	cattle; costa rica; livestock; lambs; farm animals; marlin; alpacas; dairy cows; beef cattle; poodle; pit bull; rhinos; alpaca; dairy cattle; piglets; steers; thoroughbreds; water buffalo; .
16	salmon; lion; tuna; shark; fins; catfish; jacks; slider; carp; walleye; tyrant; ide; koi; scorpion; racer; suckers; rainbow trout; steelhead; mackerel; ou; herring; sturgeon; tilapia; tarpon; big fish; pollock; halibut; eel; shad; swordfish; chinook; discus; rudd; wahoo; barracuda; flounder; atlantic salmon; anchovies; redfish; sea bass; flier; snook; calamari; sailfish; haddock; grunts; lake trout; cichlids; walleyes; garibaldi; brook trout; bonefish; deep sea; moray; bream; dorado; king salmon; coho; chinook salmon; barbs; rockfish; salmonids; barra; bonita.

Table 5: Examples of clustering results from the EM algorithm, where animal instances are randomly listed.

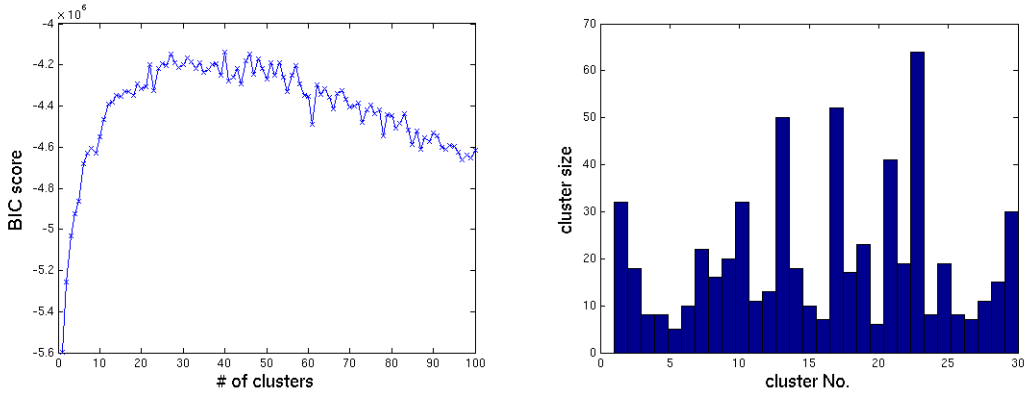

Figure 6: (left) BIC versus number of clusters using the EM algorithm, (right) Cluster size versus the cluster number using the EM algorithm when the number of clusters is fixed to be 30.

6.4 Studies with Relation Instances

As discussed in Sec. 4, we propose the CNMF method to combine side information and the original data set to improve clustering results. This has been demonstrated using synthesized data set as described in Sec. 5. To obtain side information of the animal training data set, we use the relation instances from the relation `(arg1) animal_is_type_of_animal (arg2)`. Many relation instances provide good information such as: ``gulls → seabirds'', ``bears → predators'' and ``jaguars → mammals''. Notice that, both arguments of the relation instances are from the same animal category. We can view all the animal instances as forming a simple hierarchical structure, where more general animal instances are on the top level and more specific animal instances are on the bottom. The relation instance thus provides information about how to classify more specific animals in general categories, where the first argument is most likely to be bottom nodes and the second argument is most likely to be top nodes.

For animal instances in our training data set, we retrieve 270 relation instances that have both arguments in our training data set. The full list of relation instance can be seen in Appendix. 8.1. Of these 270 relation instances, there are 134 distinct animal instances in the first argument and 63 distinct animal instances in the second argument. Taking the 63 distinct animal instances from the second argument as the feature space, we obtain a side information matrix of 63 rows and 134 columns, where the cell is one if there is a corresponding relation instance and zero otherwise. This matrix can be used directly as matrix Y in the CNMF, which provides side information for 134 out of 600 animal instances. The quality of the side information can be seen by applying NMF method using matrix Y alone. The clustering results of these 134 instances are shown in Tab. 6, where the number of clusters are set to be 12. According to this result, one can see that the side information have a reasonably good quality.

6.5 Clustering Results from CNMF

We first apply the NMF method on the original data set without side information. The clustering results are shown in Tab. 7, which are much better than previous results from K-means++ and EM algorithm based on our knowledge about animal categories. Hence we start with the NMF method, and implement the CNMF method to incorporate side information to improve clustering performance. For feature selection of the original training data, we use extraction patterns of the animal category as they give better cluster results in general than the other two feature selection methods.

To choose the value for the trade-off parameter λ of Eqn. 9, we first apply CNMF method on a hold-out data set of 100 animal instances. The hold-out data is obtained by choosing from 601st to 700th popularity ranked animal instances. For different values of the trade-off parameter, we apply the CNMF method to the training data set and check the function value of the hold-out data using the corresponding basis matrix. The function value versus different λ value is shown in Fig. 7,

Table 6: Cluster Results of relation instances (side information only) with NMF method

Class	Animal instances
1	snakes; frogs; turtles; reptiles; lizards; crocodiles; alligators; tortoises; iguanas;
2	cats; pet; puppies; wolves; bulls; donkeys; canines; carnivores; pugs;
3	waterfowl; swans; gulls; shorebirds; herons; loons; teal; cormorants; egrets; stingrays;
4	spider; kittens; leopard; mussels; cougars; kitties; leopards; felines; cheetahs; macaques;
5	horses; cattle; sheep; deer; cows; pigs; rabbits; goats; elk; camels; bison; sparrows; finches; wildebeest;
6	rats; squirrels; crickets; hamsters; scorpions; gophers; lemmings; voles;
7	cow; butterflies; pests; rodents; predator; termites; moths; hummingbirds; locusts; dragonflies; ladybugs;
8	humans; mice; mammals; bats; shrimp; fishes; larvae; crustaceans; minnows; grubs; copepods;
9	turkey; chickens; sharks; ducks; poultry; geese; turkeys; hogs; carp; kangaroos; kangaroo; crappie; amazons; shad; koalas; macaws; wallabies;
10	dogs; bears; lions; elephants; tigers; dragons; foxes; lynx; raccoons; hyenas;
11	insects; bear; livestock; buffalo; dolphins; owls; squirrel; ferrets; falcons; antelope; caribou; badgers; otters; manatees; jaguars; cetaceans; mink; skunks; hedgehogs;
12	man; monkeys; apes; chimpanzees; chimps; baboons;

where we have set rank $k = 30$ in the factorization. Notice that when $\lambda = 0$, it is equivalent to the original NMF method without side information. Our first point corresponds to $\lambda = 10^{-4}$, which is approximately the same as using NMF only. According to the results, as we increase the size of trade-off parameter, the performance improves at first, then it reaches the minimum and become worse. As a result, the hold-out data set is best factorized when $\lambda = 0.4$. We thus set $\lambda = 0.4$ and use the factorized encoding matrix to cluster the training data set. we assign class label to each animal instance such that it has the largest membership on the corresponding basis axis. The final clustering results are listed in Tab. 8.

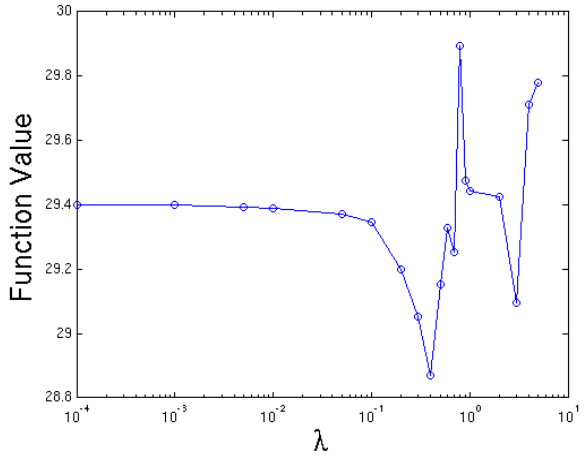

Figure 7: Function value of Eqn. 10 of 100 hold-out animal instances versus different values of the trade-off parameter λ, where for each λ value the basis matrix in Eqn. 10 is fixed to CNMF result of the training data.

7 Conclusion

We have presented the coupled non-negative matrix factorization (CNMF) method to incorporate side information on part of the data set and improve the original clustering problem on the full data set. The side information could be in the form of class labels or more general measurements in a new feature space. The CNMF method also allows a trade-off parameter to adjust the weight of the side information and we give a cross validation method to choose the value of the trade-off parameter. Using the synthesized data we have demonstrated the correctness and effectiveness of the CNMF method.

We further successfully apply the CNMF method on the real world data set and find sub-categories of noun phrases in the CMU NELL's knowledge base. According to the cross validation results from hold out data set as shown in Fig. 7, the CNMF method with a trade-off parameter $\lambda = 0.4$ achieves better factorization than the NMF method where $\lambda \approx 0$. The quality of the side information used in the CNMF has been shown in Tab. 6, where we can see that instances with side information can be nicely clustered into 12 classes. Given reasonably good side information and the cross validation results on the hold out data set, we believe that the final clustering results with CNMF is better than the results with NMF. On the other hand, it seems that results from NMF in Tab. 7 are already quite good, and it is hard to see significant improvements from results using CNMF shown in Tab. 8. One reason is that information from extraction patterns which we used as the feature selection overlaps from the side information extracted using relation instances. In other words, the extraction patterns learned by NELL already use part of the side information from the relation instances. So a feature work of this project will be to better understand this correlation and to find other good side information.

References

[1] Carlson, J. Betteridge, et al., "Architecture for Never-Ending Language Learning", Proceedings of the Conference on Artificial Intelligence (AAAI), 2010.

[2] Sneath, P. H. A, and Sokal, R. R, 1973. "Numerical Taxonomy:the Principle and practice of numerical classification", Freeman, pp.573, 1973.

[3] King, B, "Step-wise clustering procedures", J. Am. Stat. Assoc. 69, 86, 1967.

[4] Mcqueen, J, "Some methods for classification and analysis of multivariate observations", In Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, 281,1967.

[5] Dempster, A. P, Laird, N. M, and Rubin, D. B, "Maximum likelihood from incomplete data via the EM algorithm", J. Royal Stat. Soc. B. 39, 1, 1-38,1977.

[6] Arthur, D. and Vassilvitskii, S. (2007). "k-means++: the advantages of careful seeding". Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms. pp. 1027

[7] K. Beyer, J. Goldstein, R. Ramakrishnan, and U. Shaft, "When is nearest neighbor meaningful", in Proc. 7th Int. Conf. Database Theory, 1999.

[8] Pearson, K., "On Lines and Planes of Closest Fit to Systems of Points in Space", Philosophical Magazine 2 (11): 559, 1901.

[9] R. Duda, P. Hart, and D. Stork, "Pattern Classification", 2nd ed. New York: Wiley, 2001.

[10] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering", in Advances in Neural Information Processing Systems, 2002.

[11] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization". Nature, 401:788,1999.

[12] S. Arora, R. Ge, R. Kannan, and A. Moitra, "Computing a Nonnegative Matrix Factorization —Provably", ACM STOC, 2012.

[13] D. D. Lee and H. S. Seung, "Algorithms for Non-negative Matrix Factorization". Advances in Neural Information Processing Systems 13: Proceedings of the 2000 Conference. MIT Press. pp. 556, 2001.

[14] C. Lin,"Projected Gradient Methods for Non-negative Matrix Factorization" (PDF). Neural Computation 19 (10),2007.

[15] H. Kim and H. Park,"Nonnegative Matrix Factorization Based on Alternating Nonnegativity Constrained Least Squares and Active Set Method". SIAM Journal on Matrix Analysis and Applications 30 (2): 713, 2008.

[16] J. Kim and H. Park, "Fast Nonnegative Matrix Factorization: An Active-set-like Method and Comparisons". SIAM Journal on Scientific Computing 33 (6): 3261, 2011.

[17] W. Xu, X. Liu, and Y. Gong, "Document clustering based on non-negative matrix factorization". In Proc. SIGIR pp.267-273, 2003.

[18] Y. Chen, M. Rege, M. Dong, and J. Hua, "Non-negative matrix factorization for semi-supervised data clustering", Knowl. Inform. Syst., Vol. 17, pp. 355-379, 2008.

[19] L. Jing, J. Yu, T. Zeng, and Y. Zhu, "Semi-supervised Clustering via Constrained Symmetric Non-negative Matrix Factorization", Brain Informatics, Lecture Notes in Computer Science Volume 7670, pp. 309-319, 2012.

[20] H. Lee, J. Yoo, and S. Choi, "Semi-supervised Nonnegative Matrix Factorization", IEEE Signal Processing Letters, Vol. 17, No. 1, pp. 4-7, 2010.

[21] G. Salton and M.J. McGill, "Introduction to Modern Information Retrieval", McGrawHill, 1983.

[22]A. Carlson, J. Betteridge, E.R. Hruschka Jr. and T.M. Mitchell, "Coupling Semi-Supervised Learning of Categories and Relations", In Proceedings of the NAACL HLT Workshop on Semi-supervised Learning for Natural Language Processing, 2009.

Table 7: Cluster Results of 600 Animal Instances in NELL's Knowledge Base with NMF Method, where the classes are sorted according to the order of their first found instance.

Class	Animal instances
1	bats; flock; hens; crows; seagull; rook;
2	frog; chicks; cuckoo; lizard; crickets; sparrows; cicadas; wren; two birds; songbird; wild birds;
3	tiger; tigers; monkey; panthers; chimpanzees; wildcats; gorilla; polar bear; chimps; kangaroo; giraffes; rhinos; baboons; gibbons; zebras; roosters; llama; rhinoceros; grizzly; heifer; lioness; macaques; wild boar;
4	mammals; frogs; microorganisms; rodents; reptiles; predator; vertebrates; amphibians; lizards; two cats; guide dogs; service animals;
5	crab; crabs; lobsters; prawns; shrimps; krill; crawfish; copepods;
6	cattle; buffalo; elephant; elephants; elk; bison; reindeer; antelope; caribou; zebra; giraffe; llamas; herbivores; wildebeest; impala; buffaloes; gazelle; elks;
7	waterfowl; bird species; seabirds; shorebirds; songbirds; waders; waterbirds; bird populations;
8	snakes; turtles; turtle; crocodiles; tortoise; tortoises; gecko; iguanas; rattlesnake; vipers; rattlesnakes; anaconda; boas;
9	shrimp; lobster; oysters; shellfish; mussels; clams; scallops; oyster; clam; abalone; crabmeat; mussel;
10	walleye; rainbow trout; steelhead; big fish; game fish; stripers; chinook; redfish; lake trout; walleyes; brook trout; deep sea; bream; king salmon; coho; chinook salmon; salmonids; barra; smaller fish; bonita;
11	lion; wolf; wolves; lions; sharks; leopard; hawks; foxes; cougars; marine mammals; coyote; lynx; raccoons; cheetah; jaguars; bobcats; cougar; leopards; puma; cheetahs; killer whales; skunks; hyenas; mountain lion; mountain lions; family dog; possums; big cats; larger fish; peregrine;
12	eagle; rails; cardinals; owls; pigeons; merlin; penguin; falcons; raptors; hummingbirds; swift; gulls; bald eagle; orioles; herons; emu; thrush; heron; seagulls; robins; canaries; osprey; swallows; finches; amazons; dodo; woodpeckers; tits; condor; woodpecker; jays; vulture; peacocks; bluebirds; albatross; starlings; kaka; ospreys; blue jays; cormorants; macaws; moa; terns; storks; magpies; blackbirds; condors; ostriches; lovebirds; puffins;
13	larvae; caterpillars; nymphs; larva; maggots; tadpoles; minnows; grubs; nymph; minnow;
14	bees; butterfly; moth; wasp; honeybees; honey bees; pollinators; grasshopper; ladybugs; bumblebee; fire ants; fruit fly; honey bee; monarch butterfly;
15	organisms; pests; fishes; older children; invertebrates; crustaceans; arthropods; rodent; zooplankton; sea creatures; marine animals; freshwater fish;
16	bugs; butterflies; flies; ants; spiders; fleas; ticks; mosquito; beetles; termites; aphids; moths; mites; wasps; locusts; hornets; dragonflies; firefly; fruit flies; bed bugs; fireflies; mosquitos; roaches; thrips; hoppers; drosophila melanogaster; bedbugs; gnats; lepidoptera; midges;
17	dinosaurs; dragons; rays; sea turtles; hounds; jellyfish; scorpion; scorpions; werewolf; hound; starfish; unicorns; salamanders; werewolves; scavengers; nautilus; seahorses; salamander; whale sharks; stingrays; stags; baboon;
18	ducks; geese; parrots; cranes; swans; doves; pelicans; flamingos; loons; canada geese; ibis; egrets; sandhill cranes; snow geese;
19	children; man; child; animals; baby; babies; chicken; mouse; pet; prisoners; fly; bear; bass; cow; worms; lamb; bears; snake; whale; spider; kittens; kitten; cubs; preschoolers; owl; fisher; dolphin; children ages; pike; beetle; slider; marine life; perch; hare; tyrant; panda; jig; racer; pulitzer prize; mouse button; ou; hummingbird; crappie; kitties; toads; ruminants; hornet; school age children; tots; crocs; hyena; flier; little dog; awarding; recluse; prestigious award; sea turtle; angelfish; newt; tropical fish; family pets; computer mouse; chipmunk;
20	salmon; trout; tuna; shark; cod; fins; catfish; jacks; goldfish; carp; sardines; koi; suckers; snapper; herring; sturgeon; eels; pollock; eel; shad; swordfish; rudd; wahoo; barracuda; flounder; atlantic salmon; anchovies; sea bass; mullet; doubleheader; zebrafish; grunt; calamari; sailfish; haddock; grunts; cichlids; moray; basses; barbs; rockfish;
21	humans; dolphins; whales; seals; primates; apes; jaguar; beaver; roe; carnivores; sea lions; manatees; cetaceans; raccoon; orcas; mink; bobcat; walrus; mammoths; hominids; grizzly bear; bottlenose dolphins; three dogs; marsupials; armadillos;
22	dog; dogs; cat; cats; costa rica; sheep; livestock; puppy; puppies; chickens; rabbit; poultry; geek; bulldogs; farm animals; hen; bunnies; canines; companion animals; pit bulls; fowl; felines; beef cattle; poodle; pit bull; alpaca; dairy cattle; pugs; chihuahuas; poodles; pooch; canary; thoroughbreds; equines;
23	turkey; pigs; pig; turkeys; ox; boar; swine; hogs; hog; yak;
24	insects; corals; drosophila; snails; octopus; sponges; squid; slugs; nematodes; earthworms; molluscs; barnacles; earthworm; various species; sea urchins; trilobites; tarantulas;
25	roller; whitehead; teal; mallards; mallard;
26	horses; cows; goats; bulls; beasts; calves; lambs; donkey; camels; colts; ponies; oxen; mules; donkeys; alpacas; mustangs; stallions; asses; dairy cows; piglets; heifers; ewes; steers; older dogs; water buffalo;
27	duck; dove; quail; pheasant; pheasants; grouse; mantis; parakeets; warbler;
28	deer; monkeys; axis; penguins; herds; rhino; alligators; badgers; dear; kangaroos; sea life; wild animals; otters; endangered species; pandas; gators; hippo; road runner; hippos; otter; black bear; koalas; koala; wallabies; snow leopard; black bears; golden eagles;
29	creatures; creature; ide; marlin; mackerel; tilapia; tarpon; halibut; discus; snook; garibaldi; bonefish; dorado;
30	mice; rats; rabbits; squirrels; squirrel; ferrets; animal models; guinea pigs; laboratory animals; hamsters; gophers; lemmings; hedgehogs; prairie dogs; porcupine; chinchilla; voles; lab animals;

Table 8: Cluster Results of 600 Animal Instances in NELL's Knowledge Base with CNMF Method, where the classes are sorted according to the order of their first found instance.

Class	Animal instances
1	rays; dolphin; roller; jacks; squid; snapper; big fish; game fish; whitehead; stripers; chinook; rudd; wahoo; flounder; redfish; sea bass; grunt; calamari; sailfish; walleyes; bream; coho; chinook salmon; barra; smaller fish; larger fish; bonita;
2	lobster; oysters; shellfish; mussels; clams; scallops; oyster; clam; abalone; crabmeat; mussel;
3	snakes; turtles; turtle; crocodiles; alligators; endangered species; tortoises; gecko; iguanas; rattlesnake; vipers; rattlesnakes; anaconda; chinchilla; boas;
4	dog; dogs; horses; cats; costa rica; sheep; livestock; puppy; chickens; rabbit; duck; pests; goats; poultry; geek; bulldogs; farm animals; hen; bunnies; pit bulls; fowl; felines; poodle; pit bull; alpaca; chihuahuas; pooch; canary; thoroughbreds; equines;
5	geese; swans; flamingos; loons; canada geese; teal; mallards; egrets; sandhill cranes; mallard; snow geese;
6	kittens; flock; crows; seagulls; kitties; peacocks; seagull; rook; magpies; blackbirds; warbler;
7	pet; puppies; bulls; beasts; donkey; colts; ponies; oxen; mules; canines; carnivores; mustangs; stallions; asses; pugs; llama; buffaloes; stags; water buffalo;
8	children; child; animals; baby; cat; babies; chicken; prisoners; fly; bass; worms; lamb; snake; trout; whale; butterfly; kitten; frog; preschoolers; owl; fisher; children ages; pike; swine; rhino; beetle; slider; marine life; perch; hare; tyrant; panda; jig; racer; pulitzer prize; sea life; wild animals; ou; hummingbird; pandas; heron; toads; hornet; school age children; tots; crocs; jays; vulture; doubleheader; hyena; flier; little dog; awarding; recluse; ibis; prestigious award; sea turtle; angelfish; mantis; newt; lioness; tropical fish; chipmunk;
9	bugs; butterflies; flies; ants; spiders; fleas; ticks; mosquito; beetles; termites; aphids; moths; mites; moth; wasps; locusts; hornets; dragonflies; firefly; fruit flies; bed bugs; fireflies; mosquitos; roaches; grubs; thrips; hoppers; drosophila melanogaster; bedbugs; gnats; lepidoptera; midges;
10	organisms; mammals; bats; frogs; fishes; rodents; reptiles; older children; predator; invertebrates; vertebrates; amphibians; lizards; crustaceans; arthropods; rodent; zooplankton; marine animals; freshwater fish;
11	eagle; rails; cardinals; dove; parrots; pigeons; cranes; merlin; penguin; falcons; hummingbirds; swift; doves; gulls; quail; bald eagle; sparrows; orioles; pelicans; herons; emu; thrush; robins; canaries; osprey; swallows; pheasant; finches; amazons; dodo; woodpeckers; tits; pheasants; condor; grouse; bluebirds; albatross; starlings; kaka; ospreys; blue jays; cormorants; macaws; moa; terns; storks; parakeets; condors; ostriches; lovebirds; puffins;
12	walleye; rainbow trout; steelhead; crappie; sturgeon; snook; lake trout; brook trout; deep sea; king salmon; salmonids;
13	man; turkey; pigs; pig; ducks; turkeys; ox; boar; hogs; donkeys; hog; yak;
14	waterfowl; bird species; raptors; seabirds; shorebirds; songbirds; waders; waterbirds; bird populations;
15	deer; dragons; hounds; jellyfish; scorpion; werewolf; hound; starfish; unicorns; salamanders; werewolves; scavengers; nautilus; seahorses; salamander; whale sharks; baboon;
16	cattle; buffalo; elephant; elephants; elk; camels; bison; reindeer; antelope; caribou; zebra; giraffe; herbivores; zebras; wildebeest; impala; gazelle; elks;
17	insects; cow; spider; corals; drosophila; snails; octopus; sponges; slugs; nematodes; earthworms; scorpions; molluscs; earthworm; sea urchins; trilobites; tarantulas;
18	shrimp; crab; crabs; lobsters; prawns; shrimps; minnows; krill; crawfish; barnacles; copepods;
19	mouse; mouse button; computer mouse;
20	larvae; caterpillars; nymphs; larva; maggots; tadpoles; nymph; minnow;
21	dinosaurs; microorganisms; primates; tortoise; companion animals; gorilla; ruminants; two cats; poodles; guide dogs; hominids; hyenas; service animals; family pets;
22	salmon; sharks; tuna; shark; cod; fins; catfish; goldfish; carp; sardines; marlin; koi; suckers; herring; eels; tilapia; pollock; eel; shad; swordfish; discus; barracuda; atlantic salmon; anchovies; mullet; zebrafish; haddock; grunts; cichlids; moray; basses; barbs; rockfish; stingrays;
23	cuckoo; lizard; sea turtles; crickets; woodpecker; cicadas; wren; two birds; songbird; wild birds; golden eagles;
24	tiger; monkeys; tigers; monkey; penguins; apes; panthers; chimpanzees; herds; wildcats; dear; kangaroos; chimps; gators; kangaroo; hippo; road runner; giraffes; hippos; rhinos; baboons; gibbons; rhinoceros; koalas; grizzly; koala; wallabies; snow leopard; macaques; black bears;
25	mice; rats; rabbits; squirrels; guinea pigs; hamsters; gophers; lemmings; prairie dogs; porcupine; voles;
26	bees; wasp; honeybees; honey bees; pollinators; grasshopper; ladybugs; bumblebee; fire ants; fruit fly; honey bee; monarch butterfly;
27	lion; wolf; lions; leopard; hawks; foxes; cougars; marine mammals; lynx; cheetah; bobcats; leopards; cheetahs; mountain lion; mountain lions; family dog; big cats; peregrine;
28	axis; cubs; calves; chicks; lambs; hens; animal models; laboratory animals; alpacas; llamas; dairy cows; beef cattle; dairy cattle; piglets; heifers; ewes; roosters; steers; heifer; older dogs; lab animals;
29	creatures; creature; ide; mackerel; tarpon; halibut; garibaldi; bonefish; dorado;
30	humans; bear; cows; bears; dolphins; whales; wolves; seals; owls; squirrel; ferrets; jaguar; beaver; coyote; roe; raccoons; badgers; polar bear; sea lions; otters; manatees; jaguars; cetaceans; raccoon; cougar; puma; orcas; otter; sea creatures; black bear; killer whales; mink; bobcat; skunks; walrus; hedgehogs; mammoths; grizzly bear; bottlenose dolphins; three dogs; various species; wild boar; possums; marsupials; armadillos;

8 Appendix

8.1 The List of Relation Instances

dogs → child, cats, mammals, puppies, canines, carnivores, pet, children, wolves, beasts, humans, scavengers, | poultry → fowl, livestock, | herons → shorebirds, waterbirds, | ladybugs → bugs, insects, | loons → waterfowl, waterbirds, | predator → insects, | puppies → dogs, | grubs → invertebrates, | amazons → parrots, | wallabies → marsupials, | kangaroos → marsupials, | lynx → carnivores, | cormorants → seabirds, waterbirds, | sparrows → songbirds, | leopards → cats, beasts, | spider → recluse, | deer → pests, herbivores, beasts, ruminants, | wildebeest → herbivores, | carnivores → cats, dogs, | mussels → molluscs, | elk → mammals, herbivores, | hedgehogs → mammals, | chimpanzees → apes, primates, | snakes → vertebrates, reptiles, | lizards → vertebrates, reptiles, | crocodiles → reptiles, beasts, | skunks → mammals, | otters → mammals, | falcons → raptors, | crickets → pests, arthropods, | egrets → shorebirds, waterfowl, waterbirds, | frogs → vertebrates, reptiles, | canines → dogs, | dolphins → mammals, cetaceans, | pugs → dogs, | tortoises → reptiles, | hummingbirds → pollinators, | mink → mammals, | elephants → mammals, beasts, | ferrets → mammals, | bats → vertebrates, mammals, organisms, pollinators, | hamsters → rodents, | sheep → mammals, herbivores, ruminants, livestock, | chickens → poultry, fowl, livestock, | man → horses, primates, livestock, | caribou → mammals, | moths → pollinators, | bear → mammals, | buffalo → mammals, | monkeys → mammals, apes, primates, | bison → herbivores, | wolves → mammals, dogs, carnivores, beasts, | turtles → reptiles, | livestock → mammals, | reptiles → vertebrates, | waterfowl → waterbirds, | shad → minnows, | squirrels → mammals, pests, rodents, | pet → dogs, | stingrays → rays, | squirrel → mammals, | leopard → cats, | cheetahs → cats, | cetaceans → mammals, | cow → flies, | insects → mammals, bugs, arthropods, aphids, flies, | crappie → bream, | owls → mammals, raptors, | macaques → monkeys, | mice → vertebrates, mammals, organisms, pests, insects, rodents, | pests → fleas, termites, ants, insects, rats, aphids, flies, moths, | butterflies → insects, pollinators, | carp → minnows, | fishes → organisms, | donkeys → equines, | locusts → pests, insects, | lemmings → rodents, | jaguars → mammals, | gulls → seabirds, waterbirds, | turkeys → poultry, fowl, | shorebirds → waterfowl, waterbirds, | swans → waterfowl, | hogs → livestock, | koalas → marsupials, | termites → insects, | copepods → zooplankton, | camels → ruminants, | ducks → poultry, fowl, waterfowl, livestock, | teal → waterfowl, | alligators → reptiles, | humans → vertebrates, cats, mammals, organisms, dogs, beasts, primates, livestock, | cattle → mammals, herbivores, beasts, ruminants, | kitties → cats, | larvae → organisms, | bulls → dogs, | antelope → mammals, | badgers → mammals, | gophers → pests, rodents, | felines → cats, | horses → mammals, herbivores, beasts, livestock, | minnows → shad, carp, | foxes → mammals, pests, carnivores, | manatees → mammals, | cats → mammals, dogs, rabbits, carnivores, pet, children, kitties, rodents, | kangaroo → marsupials, | iguanas → reptiles, | dragons → beasts, | pigs → mammals, pests, herbivores, rodents, livestock, | chimps → apes, primates, | kittens → cats, | rats → vertebrates, mammals, pests, rodents, | cougars → cats, carnivores, | mammals → cats, organisms, dogs, pigs, rats, | scorpions → arthropods, | rabbits → mammals, herbivores, pests, rodents, livestock, | shrimp → crustaceans, | apes → primates, | raccoons → mammals, pests, carnivores, | finches → songbirds, | cows → mammals, herbivores, | dragonflies → bugs, | goats → mammals, herbivores, ruminants, livestock, | crustaceans → organisms, | baboons → primates, | voles → mammals, rodents, | bears → mammals, carnivores, beasts, | macaws → parrots, | hyenas → carnivores, scavengers, | turkey → poultry, | tigers → mammals, cats, carnivores, beasts, felines, | lions → mammals, cats, carnivores, beasts, | geese → fowl, waterfowl, | sharks → fishes, | rodents → vertebrates, mammals, pests, insects, |