Exploring Extensive Linguistic Feature Sets in Near-Synonym Lexical Choice

Mari-Sanna Paukkeri[1], Jaakko Väyrynen[1], and Antti Arppe[2]

[1] Aalto University School of Science, P.O. Box 15400, FI-00076 Aalto, Finland
[2] University of Helsinki, Unioninkatu 40 A, FI-00014 University of Helsinki, Finland

Abstract. In the near-synonym lexical choice task, the best alternative out of a set of near-synonyms is selected to fill a lexical gap in a text. We experiment on an approach of an extensive set, over 650, linguistic features to represent the context of a word, and a range of machine learning approaches in the lexical choice task. We extend previous work by experimenting with unsupervised and semi-supervised methods, and use automatic feature selection to cope with the problems arising from the rich feature set. It is natural to think that linguistic analysis of the word context would yield almost perfect performance in the task but we show that too many features, even linguistic, introduce noise and make the task difficult for unsupervised and semi-supervised methods. We also show that purely syntactic features play the biggest role in the performance, but also certain semantic and morphological features are needed.

Keywords: Near-synonym lexical choice, linguistic features.

1 Introduction

In the *lexical choice* task, gaps in a text are filled with words that best fit the context. Lexical choice is needed in many natural language generation (NLG) applications: for example, in machine translation, question-answering, summarisation, text simplification, and adapting terminology so that it can be understood by a user. It can also help produce more readable language and expand the limits of bilingual dictionaries by taking the context better into account. Further, a second-language student or translator would benefit from an application which could help write text in a foreign language by suggesting appropriate alternatives to words. Lexical choice is a very difficult problem within a set of near-synonyms due to fine-grained differences between the words. Some methods have been proposed for the problem in the literature [7,23].

In this paper, we use extensive linguistic analysis of word context in the near-synonym lexical choice task. We apply the *amph* data set [2] which contains occurrences of four *think* lexemes in Finnish with over 650 morphological, semantic, syntactic, and extra-linguistic features. It has been shown that a rich manually selected feature set improves supervised classification based on polytomous logistic regression in the near-synonym lexical choice task [2]. In this work we verify the earlier results and take a step forward by using unsupervised and semi-supervised methods in the task. This direction is important for those NLP tasks in which there is not much labelled training data available. In some tasks unsupervised methods perform as well as supervised methods, or even

A. Gelbukh (Ed.): CICLing 2012, Part II, LNCS 7182, pp. 1–12, 2012.

better (e.g., [13,24]), because of their wide coverage and ability to generalise to new data. Furthermore, unsupervised methods are good in explorative research of previously unseen data and in visualising the structure of complex data. In addition, we experiment with automatic feature selection in order to find the best-representative features for the task and to find a feature set that enhances the unsupervised results.

On a larger scale, this work aims towards understanding semantics of synonymous words: We take an explorative view, use an extensive set of linguistic features, and study how different machine learning approaches are able to find the similarities and differences between near-synonyms. We also study how syntactic, semantic, and morphological features affect the results. We examine how the number and quality of the features affect the classification accuracy in the near-synonym lexical choice task. Although our experiments are conducted for a data set of only one set of words in the Finnish language, the experimental setting is general and can be conducted for other words, data sets, and languages. The linguistic analysis of the data set is partially manual, but similar analysis can be performed with existing resources.

1.1 Related Work

The problem of lexical choice has been studied in some earlier works. [8] created a lexical choice system by considering the branches of an ontology as clusters of synonyms. The clustering was performed based on manually defined dimensions of denotational, stylistic, expressive, and structural variations. [11] proposed extraction patterns to get near-synonym differences from a synonym dictionary. [7] proposed a lexical choice method that uses co-occurrence networks. The data set contained seven English near-synonym sets, such as *difficult, hard, tough* and *give, provide, offer*. Rather recently, [23] experimented with the same data set. They used latent semantic analysis with lexical-level co-occurrence in a supervised manner by applying support vector machines. Our work concerns a similar set of near-synonyms but extends the work into Finnish, a very large set of linguistic features and a variety of machine learning approaches.

Lexical choice is closely related to other tasks common in the natural language processing (NLP) community. *Lexical substitution* [15] is a task in which a word in a context is to be replaced with a synonymous word that is also suitable for the context. However, there is no predefined list of possible answers available. Lexical substitution has gained some popularity e.g., in SemEval tasks [16,18]. In the information retrieval community, a similar task is *query expansion* [22]. A more common task is *word sense disambiguation* (WSD) [20], in which the meaning of a polysemous word is selected from a set of alternatives. Due to the similarities between lexical choice and WSD, the approaches may use the same categorisation or clustering methods. Machine translation (MT) is also a large application area [1,4]. In MT, the task is often referred as *lexical selection*, where the target word is selected from a set of possible translations. Many vector space models have been evaluated in lexical choice tasks, such as the synonym part of the TOEFL language test [14,19].

The *amph* data set has previously been analysed based on statistical measures, manual feature selection and classification based on polytomous logistic regression according to the one-vs-rest, multinomial and other heuristics [2]. Arppe observed that a supervised approach, polytomous logistic regression seems to reach an accuracy rate of 60–66% of the instances. The results did not notably improve with the addition of further granularity in semantic and structural subclassification of the syntactic roles. Subsequently, [3] compared polytomous logistic regression and other supervised approaches. They concluded that there is no large difference on the accuracy rates of the tested supervised machine learning classifiers on the *amph* data set.

2 Data

The *amph* data set used in this work represents Finnish, which is part of the Uralic language family and is known for its highly rich agglutinative morphology. The *amph* data set is a collection of the four most frequent Finnish *think* lexemes: *ajatella* (*think* in English), *harkita* (*consider*), *miettiä* (*reflect*), and *pohtia* (*ponder*). It consists of 3404 occurrences that are collected from newsgroup postings and newspaper articles. The distribution of the four lexemes is given in Table 1. The most frequent lexeme is present in about 44% of all data instances and the least frequent lexeme comprises approximately 11% of the data. The data set is publicly available[1].

Table 1. *Think* lexemes and their frequencies and percentages in the *amph* data set

Lexeme	Frequency	%
1. *ajatella (think)*	1492	43.8
2. *harkita (consider)*	387	11.4
3. *miettiä (reflect)*	812	23.9
4. *pohtia (ponder)*	713	20.9
Total	**3404**	**100.0**

The *amph* data set has been morphologically and syntactically analysed with a computational implementation of functional dependency grammar for Finnish [21], with manual validation and correction. In addition, the analysis has been supplemented with semantic and structural subclassifications of syntactic arguments and the verb-chain. For further details, see [2, Sec. 2.2]. The data set consists of 216 binary atomic features and 435 binary feature combinations. Each feature has at least 24 occurrences in the data set. The atomic features consist of morphological features, syntactic argument features, features associated with words in any syntactic position, and extra-linguistic features, such as the data source and the author of the text. The combined features consist of syntactic & semantic, syntactic & phrase-structure, syntactic argument & base-form lexeme and syntactic & morphological feature combinations. Semantic features do not exist as atomic features, but are always combined with syntactic features.

In this paper, we use two original feature sets: FULL, all 651 features, and ATOMIC, atomic features only (216 features), and compare their performance to the feature set M6, which has been manually selected from the FULL feature set to be linguistically interesting. It was presented in [2, page 194, referred as Model VI]. The set contains 46

[1] `http://www.csc.fi/english/research/software/amph`

features, consisting of 10 verb-chain general morphological features, and their semantic classifications (6 combined features), 10 syntactic argument types, and their selected or collapsed subtypes (20 features). In addition to the features present in the FULL feature set, Arppe's M6 contains some feature combinations of the original features that are available in a supplementary data table THINK.data.extra. For more details about the features and the compilation of the data sets, see [2, Sec. 2.4, 3.1].

3 Methods

In this paper, the task is to select the most suitable lexeme out of a set of near-synonym alternatives for each context. The task is referred to *fill-in-the-blank* (FITB) [7,23]: in a corpus of sentences containing one of the near-synonyms, the original lexeme is removed from the sentences and the goal is to guess which of the near-synonyms is the missing word. Thus, the task reduces to a standard classification problem. In practice, we trained methods from different machine learning approaches to conduct the lexical choice and then used a labelled test data set to evaluate the classification accuracies. In addition to the two original feature sets, automatic feature selection was performed for the FULL feature set to obtain a small subset of features that contain relevant information for the task and obtain better classification accuracy.

3.1 Feature Selection

The data set used in this work contains an extensive feature set, which also includes noise, i.e., linguistic information not crucial to the task. In a previous work, [2] experimented with different manually selected feature sets. In our work, we aim to select automatically a set of features that best distinguish between the lexemes of the data set. The technique of selecting a subset of relevant features is known as *feature* or *variable selection* [9], which can help alleviate the curse of dimensionality, enhance generalisation capability, speed up the learning process and improve model interpretability. For computational reasons, it is typically not feasible to compute an exhaustive search of all possible feature subsets. A very simple heuristic algorithm, the *forward feature selection* algorithm, starts from an empty set and adds one feature at a time, choosing the feature which most improves an evaluation criterion.

3.2 Unsupervised Learning

Unsupervised learning methods do not use any labelled data about the correct clustering or categorisation but analyse the structure of the data. We discuss three unsupervised learning methods: K-means, self-organising map, and independent component analysis.

K-means is one of the best known clustering algorithms due to its efficiency and simplicity. It clusters data items into K clusters starting from a random initialisation of cluster centroids. The algorithm alternates between two steps: each data item is first assigned to its nearest cluster centroid, and then the centroids are updated as the means of the data items assigned to the clusters. Different distance measures can be used while the Euclidean distance metric is a common choice.

The *self-organising map* (SOM) [12] is an artificial neural network that is trained with unsupervised learning. The SOM fits an approximated manifold of prototype vectors to a data distribution. During training, the prototype vectors will start to approximate the data distribution, and the prototype vectors will self-organise so that neighbouring prototypes will model mutually similar data points. SOM can be used especially for explorative data analysis and data visualisation.

Both SOM and K-means are vector quantisation methods that cluster high-dimensional data in an unsupervised manner and represent the original data with few prototype vectors. The methods can also be used as simple classifiers. On the other hand, *independent component analysis* (ICA) [5] is an unsupervised feature extraction method that finds a representation of data in a new space. ICA assumes that each data item is generated as an instantaneous linear mixture of statistically independent components. There are several algorithms which can learn both the static mixing matrix and the component activities based on the observed data and the assumption of statistical independence.

3.3 Semi-supervised Learning

A semi-supervised approach used in this work is a semi-supervised version of the k-nearest-neighbours (kNN) method (see the following section). The selected learning approach is called *self-training*, in which previously classified data points are used as additional labelled data for further classifications. We used a straight-forward extension from the 1NN classifier introduced by [25].

3.4 Supervised Learning

Since unsupervised methods may not find the correct clustering accurately, we also experiment with some supervised methods. In supervised learning, labelled data are provided and the task is to predict correct labels for previously unseen data without labels. We consider three different methods: k-nearest-neighbours (kNN), feed-forward artificial neural network (ANN), and multinomial logistic regression (MNR), one form of polytomous logistic regression. Out of these three methods, kNN and MNR have been previously applied to the *amph* data set [2,3].

The *k-nearest-neighbours* method (kNN) [6] is a non-parametric learning method that classifies new data items according to those labelled data items that are most similar to the new one. The kNN method has no parameters to be learned, but the number of neighbours k and the distance measure have to be selected.

Feed-forward artificial neural network (ANN) is a parametric method that learns a nonlinear mapping from the input features to the given output labels from training data with scaled conjugate gradient (see, e.g., [10]). The network structure has an input layer, at least one hidden layer with nonlinear activation functions and a linear output layer. We use the network for classification and define a single output for each label.

Multinomial logistic regression (MNR) [17] is a linear parametric method. It learns a mapping from continuous and categorical dependent variables, usually assuming one outcome category as a default case against which the other outcomes are contrasted. The model learns weights (log-odds) for each dependent variable.

3.5 Evaluation

The performance of the methods in this work is evaluated with *accuracy*: the ratio of correctly classified data items to all items. The results of the methods depend on the selected data set and initialisation, and thus we run an n-fold cross-validation by dividing the data into n sets, taking each set separately to be a test set, and training the data with the other $n - 1$ sets. The reported average accuracies are calculated as the mean of the fold accuracies. Statistical significances are measured with the 1-sided Wilcoxon signed rank test.

The evaluation of the unsupervised clustering methods K-means and SOM require that a label is assigned to each cluster. The label of each cluster is set as the majority label among the data items in the cluster. There might be more clusters than possible labels. If accuracy were calculated for the training data, it would approach 100% when the number of clusters approaches the number of data points. However, we use separate train and test sets in cross-validation. Thus, while the number of clusters increases the accuracy gets close to the supervised 1NN classification accuracy.

4 Experiments and Results

All the reported results have been produced with 20-fold cross-validation: each test set consists of 5% of the data, i.e., 170 instances. As a baseline method we classify all test data items to the largest category, lexeme 1. The average accuracy of the baseline is 0.44, the fraction of the largest lexeme class.

4.1 Feature Selection

We applied the forward feature selection method using the kNN classifiers with $k = \{1, 3, 5, 10\}$ as the evaluation criteria for the FULL feature set. The kNN classifier was chosen because it was significantly faster to compute than an artificial neural network or multinomial logistic regression. Both the feature selection and the following classification were computed with the same data set because of the limited size of the *amph* data set. To alleviate this limitation, we used cross-validation in the evaluation criteria.

After feature selection, a kNN classifier with the corresponding number of neighbours k was applied to the reduced feature sets. The accuracy of the classification improved with the number of included features as shown in Fig. 1. The feature sets were evaluated with 20-fold cross-validation. 5NN quickly reached a plateau around 0.65–0.66 at about 40 features and we chose to use it for the automatically selected feature set FS40. It has been included in the classification experiments.

The automatically selected set FS40 contains six morphological features, two extra-linguistic features representing information about the text source, three features that mark that one of the lexemes appear earlier in the same text, and 29 syntactic features: 12 purely syntactic features, 12 syntactic features with semantic subtypes, and 5 syntactic features with a specific word and its part-of-speech. The linguistic categorisation of the first 10 features of the FS40 set is given in Table 2. As examples, the first selected feature 1, related to indirect questions, appears with lexeme 3 (*miettiä*), when thinking

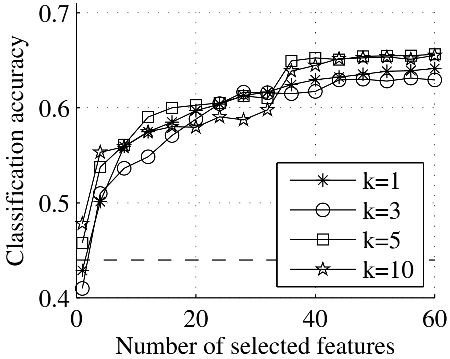

Fig. 1. Supervised classification accuracy of kNN for feature selection. The features are added incrementally with forward feature selection from the FULL feature set using kNN also in feature evaluation. The dashed horizontal line shows classification accuracy with a random classifier.

Table 2. The first ten features of the automatically selected FS40 feature set and their existence in the Arppe's M6 feature set [2]

Feature	1	2	3	4	5	6	7	8	9	10
Morphological									×	
Syntactic	×	×	×	×	×	×	×	×		×
Semantic		×					×			
PoS										×
Also in M6 [2]	×	×		×	×		×	×		

is time-limited. Feature 2 is a significant determiner of the lexeme 2 (*harkita*). Feature 3 appears with lexeme 4 (*pohtia*), and is also associated with an expression of duration for the thinking process. The first automatically selected features match with the manual analysis of features that are good at predicting and depicting the *amph* verbs [2]; 6 out of the first 10 features exist also in Arppe's M6, which is also indicated in the table. Overall, only 8 out of 40 FS40 features exist in Arppe's M6.

4.2 Unsupervised

To get an overview of the data, we first show a SOM clustering and visualisation of the FULL feature set in Fig. 2. A 10×12 SOM lattice of prototype vectors was initialised with eigenvectors corresponding to the two largest eigenvalues. The SOM was trained with the whole data set and after training the best matching cells were calculated for each data item. The labels of the data items are shown in the figure as gray-scale bars: the height of a bar corresponds to the number of data items located in each hexagon cell. As the figure shows, the lexeme selection task with the FULL feature set is very difficult for an unsupervised clustering method: the data set contains also other structure than the four lexemes, and thus SOM cannot form nicely separated clusters of the four

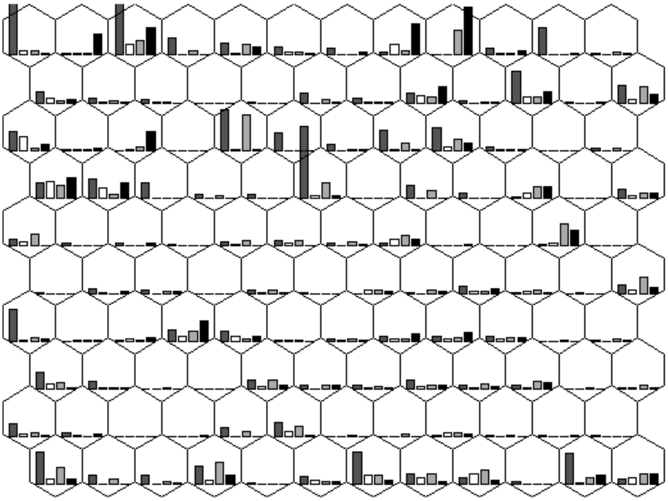

Fig. 2. Unsupervised SOM clustering and visualisation using FULL feature set. Each hexagon corresponds to one prototype vector. The grey-scale bars show the distribution of the four lexemes assigned to each cell.

lexemes. Lexeme 1 (dark grey), that occurs in about 44% of the data set, is located in the upper and left hand side part of the map. Instances of lexeme 2 (white) are in the top left corner and in the middle of the map from top to bottom. The largest occurrences of lexeme 3 (light grey) are located in the right bottom part of the map. Lexeme 3 seems to be complementary to lexeme 1. Lexeme 4 (black) is located on the top and right-hand side of the map. In the top left corner is an area of all lexemes, whereas cells with a pair of strong lexemes can be seen on many areas of the map.

Similarly to SOM, independent component analysis of the FULL feature set does not seem to extract components that match well with the *think* lexemes. The resulting components clearly find an underlying structure in the data set, but the learned structure does not reflect the wanted classification. Thus, the results are not shown here or analysed further.

The K-means classification accuracy for 20-fold cross-validation of FULL, ATOMIC, and FS40 feature sets are compared with the results of Arppe's M6 feature set in Table 3. The accuracies are calculated for the number of clusters varying between 4 and 100. The correlation distance measure was applied. FS40 needed the addition of normally distributed noise to be able to distinguish between the vectors. The automatically selected feature set FS40 performs significantly better than any of the other tested feature sets even though it contains the smallest number of features. Nevertheless, clustering into four categories does not exceed the baseline accuracy of 0.44.

Table 3. Unsupervised classification accuracy of K-means using the four feature sets. FS40 performs significantly better for all numbers of clusters K (in bold) against all other feature sets.

	FULL	ATOMIC	FS40	M6 [2]
K	**Avg**	**Avg**	**Avg**	**Avg**
4	0.44	0.44	**0.45**	0.44
6	0.44	0.44	**0.47**	0.45
8	0.44	0.44	**0.50**	0.46
10	0.44	0.45	**0.51**	0.47
20	0.46	0.48	**0.55**	0.49
30	0.49	0.48	**0.56**	0.50
50	0.52	0.50	**0.57**	0.54
100	0.54	0.51	**0.59**	0.56

4.3 Semi-supervised

Since unsupervised methods do not perform very well for the tested feature sets, we next experiment with the semi-supervised method with both labelled and unlabelled data. In the semi-supervised kNN clustering with $k = \{1, 3, 5, 10\}$ the percentages 5–100% of labelled training data were experimented. The averages of classification accuracies with the ATOMIC feature set, using 20-fold cross-validation, are shown in Fig. 3. With labelled data of 15% or more the semi-supervised 10NN performs best. With all values of k the accuracy is over the baseline when at least 15% of data is labelled. Statistically significant differences exist between 1NN and the other methods if 50% or more of the data was labelled. We got similar results also with the other feature sets (not shown).

We also tested with a fixed number of labelled data items, varying the amount of unlabelled data, and found that unlabelled data disturbs the classifier. This supports the

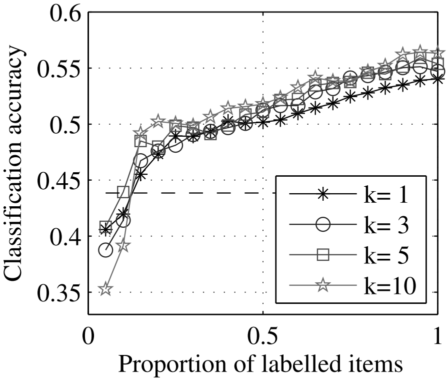

Fig. 3. Semi-supervised classification accuracy of semi-supervised kNN using ATOMIC feature set, varying the proportion of labelled data items between 0.05–1. The dashed line shows the baseline.

findings with SOM and ICA that the data set contains also some other structure than which separates the four lexemes.

4.4 Supervised

Unsupervised and semi-supervised methods were not able to find very well the structure that differentiates the four lexemes. Thus we experiment with fully labelled data. The experiments with ANN were conducted with one hidden layer of 20 neurons. The Full and Atomic feature sets were too large for MNR computation, and thus the dimensionality was reduced with principal component analysis (PCA) into 150 dimensions, which removed only a small fraction of the signal. The kNN method was run with the Euclidean distance.

Table 4 shows classification accuracy of the supervised ANN and MNR methods, and kNN with a varying number of neighbours. The feature sets are the original sets Full, Atomic, as well as the automatically selected smaller feature set Fs40. Also results with Arppe's M6 feature set [2] are shown. The averages are calculated with 20-fold cross-validation. The highest supervised accuracy, 0.66, is obtained with MNR and the Full feature set. The ANN classifier performs best with the automatically selected Fs40 and the Full set. The best results with kNN are obtained with middle values of k for all feature sets. The best result for kNN, 0.65, was obtained with the automatically selected Fs40 feature set for $k = 5$. The result is natural because the feature set was optimized for 5NN.

All the results are clearly better than the baseline 0.44. The results of Full and Fs40 are significantly better than Atomic and Arppe's manually selected M6 with the ANN classifier. For kNN, Fs40 performed significantly better than all other methods, except for the smallest value of k. In contrast, for MNR, only the Full feature set performs

Table 4. Supervised classification accuracy of ANN, MNR, and kNN with different number of neighbours k using the four feature sets. The result for the significantly best feature set is printed in bold for each method (row). For kNN, the best values of k for each feature set is underlined.

	Full	**Atomic**	**Fs40**	**M6** [2]
	Avg	**Avg**	**Avg**	**Avg**
ANN	**0.62**	0.59	**0.64**	0.59
MNR	**0.66**[1]	0.61[1]	0.60	0.63
kNN k =1	**0.60**	0.54	0.47	0.53
3	0.60	0.55	**0.64**	0.58
5	0.60	0.56	**0.65**	0.58
10	0.61	0.57	**0.63**	0.59
20	0.60	0.56	**0.64**	0.59
30	0.59	0.56	**0.63**	0.58
50	0.57	0.54	**0.62**	0.57
100	0.54	0.54	**0.61**	0.56

[1] Computed for the first 150 principal components.

better than Arppe's M6, possibly because the feature set was selected using MNR results [2]. The results show that supervised feature selection can reduce the complexity of a parametric supervised method (ANN) without lowering quality and even improve a non-parametric supervised methods (kNN) by selecting features relevant to the task.

5 Discussion and Conclusions

In this paper we have studied the use of an extensive set of linguistic features from the *amph* data set in the near-synonym lexical choice task. We used a number of machine learning methods and experimented on an automatically selected feature set. While the automatically selected feature set uses a significantly smaller number of features, the results are comparable to the original feature sets.

The best classification accuracy obtained in the task was 0.66 with multinomial logistic regression (MNR) for the FULL feature set of 651 linguistic features, by first reducing the original dimensionality with principal component analysis to 150. The automatically selected feature set FS40 of only 40 features performed overall very well: it improved over the manually selected Arppe's M6 feature set [2] with ANN and gave a comparable result to the FULL feature set. It also gave better results than any of the other feature sets with K-means and kNN. The automatically selected feature set FS40 consists mostly of syntactic features, but also some semantic and morphological features were selected as the most important ones. The FULL set generally improved over the ATOMIC set, suggesting that combining or extracting features can help classification. An analysis of the effect of different manually selected syntactic, semantic, and morphological feature sets can be found in [2, p. 207].

All tested supervised methods reached approximately the same level of performance, the best classification accuracies of each method were between 0.60 and 0.66. This supports the findings in [3] which says that this is the maximum accuracy that can be obtained with supervised methods for this data set. Unsupervised methods did not perform as well as supervised methods, which is natural behaviour with a complex data set like *amph*. However, supervised feature selection can improve unsupervised classification accuracy with an additional advantage of a significantly smaller set of features. Unsupervised feature selection based on information theoretic measures instead of supervised feature selection would be a step towards a completely unsupervised method. Feature selection based on the simple kNN classifier does not improve the results of the other supervised classifiers, when comparing to the FULL feature set. However, the smaller models contribute to faster model training as well as smaller memory and computational complexity.

Acknowledgments. We kindly thank Oskar Kohonen, Krista Lagus and Timo Honkela for their valuable comments while writing this paper. This work has been financially supported by The Finnish Graduate School in Language Studies (Langnet), Academy of Finland directly and through AIRC Centre of Excellence, and META-NET Network of Excellence.

References

1. Apidianaki, M.: Data-driven semantic analysis for multilingual WSD and lexical selection in translation. In: Proceedings of EACL 2009, pp. 77–85. ACL (2009)
2. Arppe, A.: Univariate, bivariate, and multivariate methods in corpus-based lexicography–a study of synonymy. Ph.D. thesis, University of Helsinki, Finland (2008)
3. Baayen, R.H., Arppe, A.: Statistical classification and principles of human learning. In: Proceedings of QITL, vol. 4 (2011)
4. Carpuat, M., Wu, D.: Improving statistical machine translation using word sense disambiguation. In: Proceedings of EMNLP-CoNLL 2007, pp. 61–72 (2007)
5. Comon, P.: Independent component analysis, a new concept? Signal processing 36(3), 287–314 (1994)
6. Cover, T.M., Hart, P.E.: Nearest neighbor pattern classification. IEEE Transactions on Information Theory 13(1), 21–27 (1967)
7. Edmonds, P.: Choosing the word most typical in context using a lexical co-occurrence network. In: Proceedings of EACL 1997, pp. 507–509. ACL (1997)
8. Edmonds, P., Hirst, G.: Near-synonymy and lexical choice. Computational Linguistics 28(2), 105–144 (2002)
9. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. Journal of Machine Learning Research 3, 1157–1182 (2003)
10. Haykin, S.: Neural networks: a comprehensive foundation. Prentice-Hall, Englewood Cliffs (1994)
11. Inkpen, D., Graeme, H.: Building and using a lexical knowledge base of near-synonym differences. Computational Linguistics 32(2), 223–262 (2006)
12. Kohonen, T.: Self-Organizing Maps. Springer Series in Information Sciences, vol. 30. Springer, New York (2001)
13. Kurimo, M., Creutz, M., Turunen, V.: Overview of morpho challenge in CLEF 2007. In: Working Notes of the CLEF 2007 Workshop, pp. 19–21 (2007)
14. Landauer, T.K., Dumais, S.T.: A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. Psychological Review 104(2), 211–240 (1997)
15. McCarthy, D.: Lexical substitution as a task for WSD evaluation. In: Proceedings of SIGLEX/SENSEVAL 2002, pp. 109–115. ACL (2002)
16. McCarthy, D., Navigli, R.: SemEval-2007 task 10: English lexical substitution task. In: Proceedings of SemEval 2007, pp. 48–53. ACL (2007)
17. McCullagh, P., Nelder, J.A.: Generalized Linear Models. Chapman & Hall, New York (1990)
18. Mihalcea, R., Sinha, R., McCarthy, D.: SemEval-2010 Task 2: Cross-lingual lexical substitution. In: Proceedings of SemEval 2010, pp. 9–14. ACL (2010)
19. Sahlgren, M.: The Word-Space Model. Ph.D. thesis, Department of Linguistics, Stockholm University, Stockholm, Sweden (2006)
20. Schütze, H.: Dimensions of meaning. In: Proceedings of SC 1992, pp. 787–796. IEEE (1992)
21. Tapanainen, P., Järvinen, T.: A non-projective dependency parser. In: Proceedings of Applied Natural Language Processing, pp. 64–71. ACL (1997)
22. Voorhees, E.M.: Query expansion using lexical-semantic relations. In: Proceedings of ACM SIGIR 1994, pp. 61–69. Springer, Heidelberg (1994)
23. Wang, T., Hirst, G.: Near-synonym lexical choice in latent semantic space. In: Proceedings of Coling 2010, pp. 1182–1190. ACL (2010)
24. Yarowsky, D.: Unsupervised word sense disambiguation rivaling supervised methods. In: Proceedings of ACL 1995, pp. 189–196. ACL (1995)
25. Zhu, X., Goldberg, A.B.: Introduction to semi-supervised learning. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers (2009)