Multi-label Text Classification using Semantic Features and Dimensionality Reduction with Autoencoders

Wael Alkhatib, Christoph Rensing, and Johannes Silberbauer

Technische Universität Darmstadt, Fachgebiet Multimedia Kommunikation, S3/20,
Rundeturmstr. 10, 64283 Darmstadt, Germany
{wael.alkhatib,christoph.rensing}@kom.tu-darmstadt.de
johannes_david.silberbauer@stud.tu-darmstadt.de

Abstract. Feature selection is of vital concern in text classification to reduce the high dimensionality of feature space. The wide range of statistical techniques which have been proposed for weighting and selecting features suffer from loss of semantic relationship among concepts and ignoring of dependencies and ordering between adjacent words. In this work we propose two techniques for incorporating semantics in feature selection. Furthermore, we use autoencoders to transform the features into a reduced feature space in order to analyse the performance penalty of feature extraction. Our intensive experiments, using the EUR-lex dataset, showed that semantic-based feature selection techniques significantly outperform the Bag-of-Word (BOW) frequency based feature selection method with term frequency/inverse document frequency (TF-IDF) for features weighting. In addition, after an aggressive dimensionality reduction of original features with a factor of 10, the autoencoders are still capable of producing better features compared to BOW with TF-IDF.

Keywords: semantics; feature selection; dimensionality reduction; text classification; semantic relations; autoencoders.

1 Introduction

Text classification applications have become widespread as a result of the tremendous growth in the amount of data, most of which are unstructured [1]. Popularised by search engines like Google, searching through large amounts of natural language text has become a key research topic. Since most users search for documents using concepts describing a conceptual topic, techniques based on literal word matching are often not good enough to produce relevant results [2]. The need for machine learning techniques for text classification has emerged as a result of the fact that, the amount of data to be searched and classified is too large to rely on classification by human subject matter experts outside very limited high value application fields.

Classification problems deal with the task of assigning a number of classes C out of a predefined set of classes L to an input. Such problems can either be binary, multi-class or multi label[3]. Binary classification is the problem of assigning one out of two labels meaning that $|C| = 1$ and $|L| = 2$. A problem where the task is to assign exactly one class C out of $|L|$ mutually exclusive classes to an input is called multi-class, while a classification problem is called a multi-label classification problem when the task is to classify the input into $m = |C|$ out of the set of classes L where $m \leq |L|$.

Text representation is an essential preprocessing step in text classification where documents are transformed into a format consumable by machine learning models. This involves representing each document as a vector with the size of the vocabulary where each dimension corresponds to the relevance of a concept to the document [4]. Relevance can for example be computed using weighting schema i.e. TF-IDF. In general this method produces high dimensional, sparse vectors which are extremely challenging for learning algorithms [5]. To increase the manageability of the problem, machine learning techniques apply a process called dimensionality reduction which aims at reducing redundancy and noise in the data set by mapping it into a lower dimensional space using a wide range of feature selection and extraction techniques. The potential of such techniques to improve computational efficiency and result accuracy has been demonstrated as well [6].

In this work, we propose a new method for incorporating semantic knowledge into feature selection for dimensionality reduction. Using linguistic filters we extract all noun phrases to provide a terminology of basic and extended concepts. Then we extract semantic relations between the noun phrases in order to build an acyclic directed graph as a basic shallow ontology of the documents. Using the directed graph of concepts, we propose different techniques to select the features based on the relationship between concepts. Further, aiming to a reduced feature space, we investigate the trade-off between the dimensionality reduction factor and the performance penalty using autoencoders. The empirical evaluation results showed that two of our proposed methods significantly outperform the baseline approach of BOW with TF-IDF weighting method using different multi-label classifiers.

The paper is organized as follow: An overview of related work in feature selection and extraction is provided in Sect. 2. We introduce our concept for the semantic-based feature extraction in Sect. 3. Section 4 presents the evaluation metrics while Sect. 5 demonstrates the comparative analysis and evaluation of the proposed methods against TF-IDF as a baseline. Finally, Sect. 6 summarizes the paper and discusses future work.

2 Related Work

Dimensionality reduction can be achieved by feature selection and feature extraction [7]. In the following, we introduce a variety of methods which fall into these two categories and we relate them to our methodology.

2.1 Feature Selection

Feature selection handles the problem of selecting a subset of features that is most effective for building a good predictor. This can be done by statistical or semantic-based measures [8]. The more widely used feature selection approaches are the statistical-based [9–11]. The most common methods include Information Gain (IG) and Chi-Square (Chi2). Information Gain (IG) makes use of the presence and absence of a concept in a document to select its features, while Chi2 measures the degree of dependence between a concept and a category as a base to select the features. The major drawback of the earliest statistical-based feature selection is ignoring textual features dependencies, structure and ordering.

Incorporating text semantics can provide better performance with regard to the used feature selection techniques. Masuyama et al. analysed the impact of selecting terms as features based on their part-of-speech (POS) specifically nouns, verbs, adjectives and adverbs. By analysing the different combinations of these four categories, they found out that a much smaller feature set of nouns is able to perform better than other POS combined [12]. D.D. Lewis used all noun phrases that occurred at least twice as feature phrases in text categorization [13]. After applying clustering of phrases and words, he concluded that phrases produce less effective representation than single words. Y. Liu et al. showed that using bi-gram and tri-gram to leverage context information of word depending on previous or next words can improve the performance, however, word sequence of more than 3 decreases the performance [14, 15]. A. Khan et al used frequent sequence (MSF) for extracting of associated frequent sentences and co-occurring terms. Also, they used WordNet [16], a lexical database, as a domain ontology to convert these terms to concepts and update the SVM with new feature weights [17] which also leads to a better performance. Other researchers incorporate the ontological knowledge for training-less ontology-based text classification or to provide meta-information for feature selection [18–20].

2.2 Feature Extraction

Feature extraction attempts to build a new optimised set of features from the original dataset i.e. the text documents or the selected features. One of the most widely used and well known statistical-based methods for reducing the dimensionality is principal component analysis (PCA) [21]. The aim of this technique is to find the directions of greatest variance in the data. The data set is then represented as a linear combination of those directions. This presumes that the data is located in a low-dimensional linear space and discards class information [22]. Similar to PCA, linear discriminant analysis (LDA) tries to find a linear combination of variables to represent the data but takes class assignment information into account. Another technique relying on linear combinations is local linear embedding (LLE) [23]. This technique attempts to represent each data point through a linear combination of its neighbours. A further technique introduced by L. Maaten et al. t-distributed stochastic neighbour embedding (t-SNE) is particularly useful for reducing the feature space to two or three dimensions

for visualisation [24]. It strives to preserve similarity between data points and has been successfully applied not only to documents [25] but also to other fields like malicious software [26]. The algorithm scales quadratically with the number of samples making the technique computationally expensive [27]. There are a multitude of extensions and variations to the algorithms described above as well as further different approaches [28–30].

Previously, researchers have incorporated text semantics in feature selection by selecting noun phrases or n-grams as features, others tried to leverage external lexical databases mainly WodNet to enhance the performance more. However, extracting ontological associations using external lexical resources has shortcomings due to the small coverage of concepts for particular domains and thus less ontological entities can be acquired. In our work, we improve on previous research by considering words context and dependencies to extract single and multi-word noun phrases as candidate features. Later on, instead of relying on external thesaurus, we extract semantic associations between concept pairs from the unstructured text using lexico syntactic patterns. Finally, based on the semantic relations between concepts in the taxonomic hierarchy of relationships, we propose four methods for selecting features based on semantics. Moreover, we analyse the performance penalty of using autoencoders for constructing reduced feature space from the original feature set.

3 Methodology

In the proposed method, we incorporate text semantics by taking context information and dependencies of words in consideration to select new features. Later on, we analyse the trade-off between dimensionality reduction factor and performance penalty using autoencoders. As shown in Fig. 1, the proposed approach consists of the following steps:

3.1 Linguistic Filter

In the first step we identify the domain terminology by extracting all noun phrases in order to form the basis for our semantic relation extraction phase. The role of the linguistic filter is to recognize essential concepts and filter out sequence of words that are unlikely to be concepts using linguistic information. In the linguistic component, the text documents need to be preprocessed by a part-of-speech tagger for marking up the words in a text (corpus), based on their context, as corresponding to a particular part of speech i.e. noun, preposition, verb, etc. Multi-word NP like Supervised Machine Learning will be considered as one feature and concatenated as supervised_machine _learning. A combination of 3 linguistic filters is used to extract multi-word noun phrases NPs that can reflect essential concepts.

- $Noun\ Noun+$
- $Adj\ Noun+$
- $(Adj|\ Noun)+\ Noun$

3.2 Stop-word Removal

In this phase, words that are unlikely to be part of concepts are excluded using stop-words list. A stop-word is a word that frequently appears with no strong association to a particular domain terminology and thus it is not expected to occur as concept word i.e. "regularly", "followed", "mostly", "everywhere", etc.

3.3 Semantic Relation Extraction

The aim here is to identify noun phrases which represent a concept or an instance of a concept, through extracting both explicit and implicit semantic relations i.e. Hypernym *(Is-A)* or Meronymy *(Part-Whole)* from all documents in the used corpus. In this work we will extract only taxonomic relations which are main components for building the concepts hierarchy. A taxonomy is an acyclic directed graph representing the is-a relationship between concepts in an ontology. For building the taxonomy, we use lexico syntactic pattern-based approach, specifically we use Hearst [31] six patterns for taxonomic relations. We choose the pattern-based approach due to its high precision compared to other linguistic or statical approaches. However, these patterns suffer from low recall also cover a small portion of the semantic relations in the corpus since they rely on the explicit presence of taxonomic relations between concepts. The used patterns are as follow:

- NP *such as* $\{NP,\}* \{(or|\ and)\}\ NP$
- *sush* NP *as* $\{NP,\}* \{(or|\ and)\}\ NP$
- $NP\ \{,NP\}*\{,\}$ *or other* NP
- $NP\ \{,NP\}*\{,\}$ *and other* NP
- $NP\ \{,\}$ *including* $\{NP,\}* \{(or|\ and)\}\ NP$
- $NP\ \{,\}$ *especially* $\{NP,\}* \{(or|\ and)\}\ NP$

3.4 Semantic-based Feature Selection

We propose four different feature selection techniques based on the associations between the extracted concepts using the linguistic filter and the taxonomic relations as shown in Fig. 2. Based on the graph theory we can identify candidate features using the concept position in the hierarchy and the associated sub-concepts.

- ***Concept-Document Frequency (C-DF)***: The number of documents where this concept occurs.
- ***Associated Concepts***: The number of sub-concepts underneath in the taxonomic hierarchy.
- ***Concept Height***: The degree is the number of edges connected to the concept, in other words, direct sub-concepts.
- ***Concept Degree***: The height is the number of edges on the longest downward path between that concept and a sub-concept.

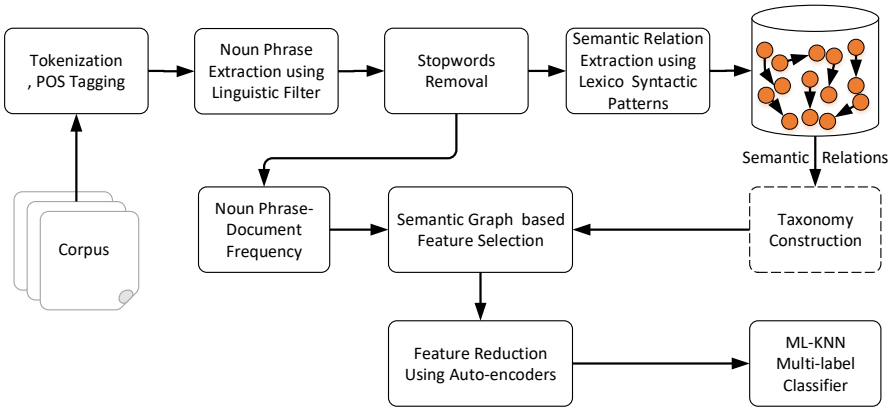

Fig. 1. Block diagram of the proposed semantic-based feature selection method

3.5 Feature Transformation with Autoencoders

A basic autoencoder is a feedforward, non-recurrent neural network trained to learn a reconstruction of its input. It consists of input layer and output layer with several hidden layers in between. The key element is a bottleneck in the middle that forces the network to learn an encoded version of its data [32]. This concept is illustrated in Fig. 3. This approach has been shown to outperform linear approaches for dimensionality reduction i.e. PCA or LDA Sect. 2 as well as more recent algorithms [33].

The network can be looked upon as a two part function: One for encoding $e = E(x)$ and another for decoding to a reconstruction of the input $r = D(e)$. The network is trained to learn an approximate reconstruction of its input : $x \approx r = D(E(x))$.

3.6 ML-KNN Multi-label Classifier

Multi-label k Nearest Neighbors (ML-KNN) results from the modification of the k Nearest Neighbors (KNN) lazy learning algorithm using a Bayesian approach in order to deal with multi-label classification problems [34]. ML-KNN searches for the k nearest neighborhood of an input instance using KNN, then it calculates prior and posterior probabilities based on frequency counting of each label y in the set of labels L in order to determine the label set of the instance. This method has been selected because experiments on three different real-world multi-label learning problems showed that ML-KNN achieves superior performance to some well-established multi-label learning algorithms [34]. Also the selection of labels based on nearest neighborhood is more convenient with our feature selection technique which consider features order and dependencies.

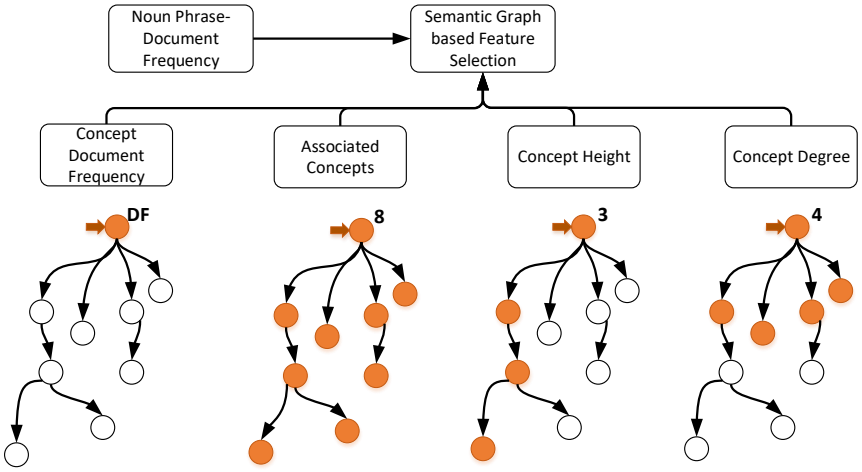

Fig. 2. Semantic feature selection based on concept associations to the underneath sub-concepts.

4 Evaluation Metrics

A classifier can either be evaluated by examining each label separately and then averaging the results. Such schemes are called *label-based*. Another approach is by considering the average difference between the expected and the predicted sets of labels over all test examples, such metrics are called *example-based*.

For a number of classifier predictions, we have the number of *true positive(TP), false positive (FP), true negative (TN)* and *false negative (FN)* predictions respectively. From those numbers we can calculate the evaluation metrics mentioned below:

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

$$F - Measure = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{3}$$

The total label-based evaluation measures for a multi-label problem where TP_j, FP_j, TN_j, FN_j are the predictions for the *j-th* label. A micro-averaged metric M_{micro} is defined as:

$$M_{micro} = M\big(\sum_{j=1}^{q} TP_j, \sum_{j=1}^{q} FP_j, \sum_{j=1}^{q} TN_j, \sum_{j=1}^{q} FN_j\big) \tag{4}$$

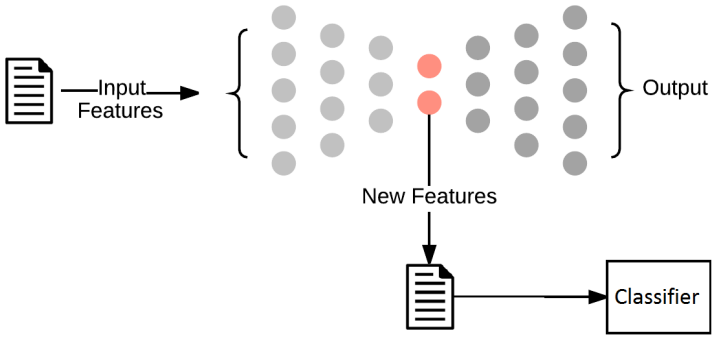

Fig. 3. Feature transformation using autoencoders

While macro-averaged metric M_{micro} is defined as:

$$M_{macro} = \frac{1}{q}\sum_{j=1}^{q} M\left(TP_j, FP_j, TN_j, FN_j\right) \tag{5}$$

In addition, one example based metric, the Hamming Loss, is used in our evaluation:

$$HammingLoss(h, D) = \frac{1}{|D|}\sum_{i=1}^{|D|}\frac{xor(Y_i, Z_i)}{|L|} \tag{6}$$

D is the set of examples (x_i, Y_i) with $Y_i \subseteq L$ and Z_i is the predicted set of labels for x_i.

5 Evaluation

In the context of our comparative analysis, the EUR-lex dataset has been used [35]. It is a text dataset containing European Union laws, treaties, international agreements, preparatory acts and other public documents. It contains 19.348 text documents, which are published in 24 official languages of the European Union. The EUR-Lex repository readily contains three different labelling schemes - *directory-codes, subject-matters* and *eurovoc-descriptors* - for its documents. However, for the evaluation we used only *subject-matters*. A detailed description of parsing and obtaining the documents, the TF-IDF features as well as the dataset properties can be found here [36]. Table 1 provides a summary of the characteristics of the subject-matters labelling scheme.

Stanford CoreNLP toolkit [37] was used in this work for performing the different natural language processing tasks (POS, linguistic filter and taxonomic

Table 1. Data-Set statistics

	Unique Labels	Label Cardinality	Label Density
Subject Matters	201	2.21	1.10

relations extraction). It combines machine learning and probabilistic approaches to NLP with sophisticated, deep linguistic modelling techniques. This toolkit provides state-of-the-art technology for wide range of natural-language processing tasks. Also it is quite widely used, both in the research NLP community, industry, and government.

The used linguistic filter to extract single and multi-word concepts resulted in 940685 distinct features. Then lexico syntactic pattern-based approach for extracting semantic relations between different features was applied using Hearst patterns for taxonomic relation extraction. Thus only taxonomic *(Is-A)* relationships were extracted, this phase resulted in 26333 is-a relationships. By incorporating other patterns from [38] in addition to Hearst patterns, we managed to retrieve 47500 relations but with significantly lower precision. For that, in the next steps we used the taxonomic relations extracted only from Hearst six patterns due to their higher precision. The extracted relations resulted in an acyclic directed graph representing the is-a relationships in the dataset.

5.1 Semantic-based Feature Selection

The carried out experiments aimed to compare the effectiveness of using semantic-based feature selection techniques against the BOW model of TF-IDF as a baseline. Four different feature selection techniques were evaluated *Concept Height*, *Concept Degree*, *Associated concepts* and *Concept-Document Frequency* with binary weighting of the features. TF-IDF with BOW feature selection was used for comparison since this approach was successfully implemented as statistical feature selection method [36]. For multi-label classification we used ML-KNN with the number of nearest neighbours $K = 10$ as fixed parameter during the experiments. In addition the number of features was fixed to 5000 features for the comparative analysis with the original TF-IDF feature set provided by EL Mencía et al. [36]. The used ML-KNN classifier for the evaluation was implemented using the MULAN open-source library for multi-label classification [39].

Figure 4 shows the cross-validation evaluation results of the different performance metrics namely, Macro/Micro-averaged F-Measure, Subset Accuracy and Average Precision. The figure compares the proposed semantic-based feature selection techniques against the baseline using ML-KNN with same configurations. Higher values indicate better performance for these metrics while lower values indicate better performance for Hamming Loss in Fig. 5. The results indicate that *Associated Concepts* and *Concept-Document Frequency (C-DF)* significantly outperformed the baseline over all performance metrics, while the baseline performed better compared to *Concept Degree* and *Concept Height*. *Associated Concepts* and *C-DF* had relative reduction in Hamming Loss of %15.38

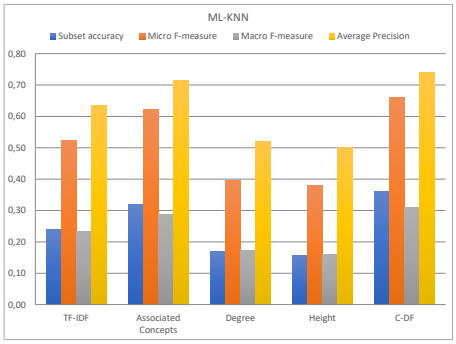

Fig. 4. Evaluation metrics of using ML-KNN with TF-IDF and BOW feature selection against different semantic-based feature selection techniques.

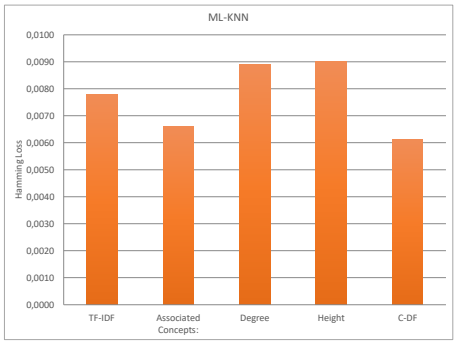

Fig. 5. Hamming Loss of using ML-KNN with TF-IDF and BOW feature selection against different semantic-based feature selection techniques.

and %21.79 respectively over TF-IDF with BOW feature selection Fig.5, which indicates lower probability of an incorrect prediction of the relevance of an example to a class label.

For more comprehensive evaluation, we compared ML-KNN using *C-DF* as feature selection techniques with a set of multi-label classifiers of the two main classifier categories namely, transformation and adaptation approaches using TF-IDF with BOW for feature selection and weighting. The used methods are Binary Relevance, Clustering Based, HOMER, BPMLL, HMC, BRKNN and Pruned Sets. We selected these methods because they have very distinct classification procedures. Figure 6 shows that ML-KNN with *C-DF* as feature selection technique had the best performance with the lowest Hamming Loss value.

The significant improvement in performance using *Associated Concepts* and *C-DF* aligns with previous researches which proved the importance of considering text semantics for feature selection. However, *C-DF* also outperformed *Associated Concepts* technique which can be justified by the relatively low number of extracted semantic relations using Hearst patterns. Based on that, further improvement is possible by integrating other techniques for associating different concepts based on their taxonomic and none-taxonomic relations. Also the lower performance of *Concept Degree* and *Concept Height* is roughly related to the low document frequency for concepts with less number of associated concepts underneath in the hierarchy.

5.2 Feature Transformation with Autoencoders

In this evaluation phase, we analysed the trade-off between dimensionality reduction factor and performance penalty. The input for the autoencoders network were the top 5000 *C-DF* features. Also ML-KNN was used with the number of nearest neighbours $K = 10$.

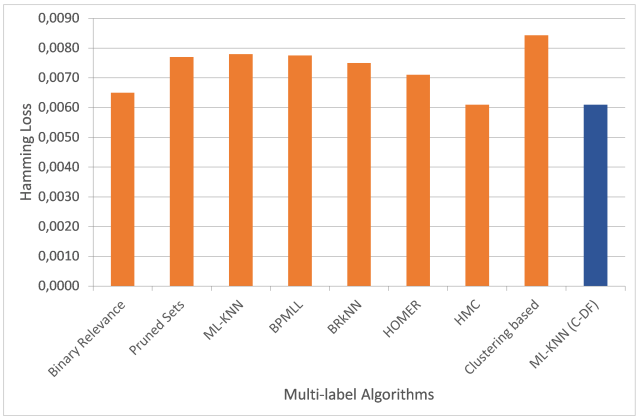

Fig. 6. Comparative analysis of ML-KNN using *C-DF* as feature selection techniques against a set of multi-label classifiers with TF-IDF features.

During the experiments, different layers configuration for the network have been evaluated as shown in Table 2. The resulted feature set of *C-DF* was directly fed into the autoencoder network. The autoencoder were configured with a network of 5 hidden layers. To analyse the effect of layer sizes on the classification results, we applied the evaluation multiple times for the different layers configurations. The outer layer size remained fixed since those have to be the size of the original *C-DF* features. We trained the network for one iteration using a batch size of 1000. The autoencoder network was implemented using the open-source Deeplearning4j framework.

Table 2. The autoencoder layers configuration for each experiment

Experiment #	Layers Configurations
1	10-250-1000-5000
2	30-500-1500-5000
3	50-800-2000-5000
4	100-800-2500-5000
5	150-800-3000-5000
6	250-800-3500-5000
7	500-800-4000-5000
8	750-2000-4500-5000
9	1000-2500-4800-5000
10	1500-2800-4800-5000
11	2000-3000-4800-5000
12	3000-3500-4800-5000

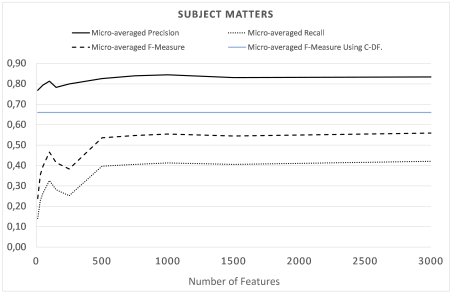

Fig. 7. Performance evaluation of ML-KNN with different number of reduced features from the original *C-DF* set using autoencoders as feature extraction technique.

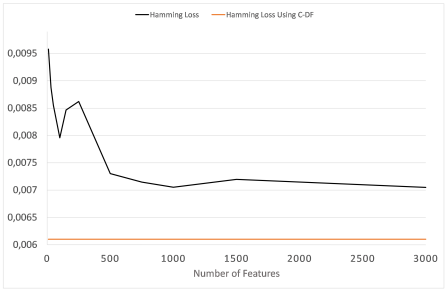

Fig. 8. Hamming Loss of ML-KNN with different number of reduced features from the original *C-DF* set using autoencoders as feature extraction technique.

Here we considered the performance of ML-KNN with *C-DF* as feature selection method with no further feature extraction as a baseline. Figure 7 shows that the classification performance based on the micro-averaged metrics, namely recall, precision and F-Measure was significantly improving when the number of encoded features increased till 500 features. However, the performance slightly improved or was almost flat with higher number of features than 500. The micro-averaged F-measure converged towards a common value of $0,56$ which still better than the performance of ML-KNN using TF-IDF with micro-averaged F-measure equals $0,52$. This means using semantic-based features with feature transformation using the autonecoder of factor 10 (transforming the original 5000 features into 500) can still provide better performance compared to the baseline statistical approach. However, Fig.7 shows a drop in the performance when the number of used features equals 250 then it increased again for 500, this give us insights about the quality of the features and that more effort should be paid on configuring the autonecoders hybrid parameters. Same conclusion can be applied on Hamming Loss Fig. 8, which changed more drastically till 500 features and then slightly improved when the number of features increased above 500.

6 Discussion

In this work we proposed four methods to select semantic-based features without relying on any external lexical databases or dictionaries. The *Associated Concepts* and *C-DF* significantly outperformed the statistical-based approach of TF-IDF with BOW feature selection over different multi-label classifiers. The different techniques proved that taking in consideration the structure, order and dependencies between words can provide better performance with regard to the statistical-based approaches. Furthermore, using autoencoders we showed that even aggressive dimensionality reduction up to a factor of 10 produced better results compared to the baseline. We also found that classification works better

for a lower number of features which agrees with other works in classification [36]. In order to balance accuracy against the increase in computation time, we identify a number of about 500 features to be the best compromise between the two dimensions. For further clarification a more thorough exploration of performance metrics on the level of individual labels need to be done. Also the proposed feature selection techniques can be improved by integrating other methods for selecting more semantic relations between the words like using bootstrapping to discover more patterns. The more patterns we have, the more information we can extract from the corpus. Furthermore, the evaluation could be widened to include other types of autoencoders, e.g. denoising autoencoders.

References

1. I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2016.
2. S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American society for information science*, vol. 41, no. 6, p. 391, 1990.
3. M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Information Processing & Management*, vol. 45, no. 4, pp. 427–437, 2009.
4. F. Sebastiani, "Text categorization," in *Encyclopedia of Database Technologies and Applications*. IGI Global, 2005, pp. 683–687.
5. I. K. Fodor, "A survey of dimension reduction techniques," *Center for Applied Scientific Computing, Lawrence Livermore National Laboratory*, vol. 9, pp. 1–18, 2002.
6. P. Cunningham, "Dimension reduction," in *Machine learning techniques for multimedia*. Springer, 2008, pp. 91–112.
7. P. Pudil and J. Novovičová, "Novel methods for feature subset selection with respect to problem knowledge," in *Feature Extraction, Construction and Selection*. Springer, 1998, pp. 101–116.
8. I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of machine learning research*, vol. 3, no. Mar, pp. 1157–1182, 2003.
9. H. Ogura, H. Amano, and M. Kondo, "Feature selection with a measure of deviations from poisson in text categorization," *Expert Systems with Applications*, vol. 36, no. 3, pp. 6826–6832, 2009.
10. P. Soucy and G. W. Mineau, "Beyond tfidf weighting for text categorization in the vector space model," in *IJCAI*, vol. 5, 2005, pp. 1130–1135.
11. Y. Yang and J. O. Pedersen, "A comparative study on feature selection in text categorization," in *Icml*, vol. 97, 1997, pp. 412–420.
12. T. Masuyama and H. Nakagawa, "Cascaded feature selection in svms text categorization," in *International Conference on Intelligent Text Processing and Computational Linguistics*. Springer, 2003, pp. 588–591.
13. D. D. Lewis, "Feature selection and feature extraction for text categorization," in *Proceedings of the workshop on Speech and Natural Language*. Association for Computational Linguistics, 1992, pp. 212–217.
14. Y. Liu, H. T. Loh, and W. F. Lu, "Deriving taxonomy from documents at sentence level," *HAD Prado and E. Ferneda," Emerging Technologies of Text Mining: Techniques and Applications", Idea, Hershey, PA*, pp. 99–119, 2007.

15. J. Fürnkranz, "A study using n-gram features for text categorization," *Austrian Research Institute for Artifical Intelligence*, vol. 3, no. 1998, pp. 1–10, 1998.
16. G. A. Miller, "Wordnet: a lexical database for english," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.
17. A. Khan, B. Baharudin, and K. Khan, "Semantic based features selection and weighting method for text classification," in *Information Technology (ITSim), 2010 International Symposium in*, vol. 2. IEEE, 2010, pp. 850–855.
18. M. Janik and K. Kochut, "Training-less ontology-based text categorization," in *Workshop on Exploiting Semantic Annotations in Information Retrieval (ESAIR 2008) at the 30th European Conference on Information Retrieval, ECIR*, vol. 20, 2008.
19. Y.-H. Chang and H.-Y. Huang, "An automatic document classifier system based on naive bayes classifier and ontology," in *Machine Learning and Cybernetics, 2008 International Conference on*, vol. 6. IEEE, 2008, pp. 3144–3149.
20. S. Chua and N. Kulathuramaiyer, "Feature selection based on semantics," in *Innovations and Advanced Techniques in Systems, Computing Sciences and Software Engineering*. Springer, 2008, pp. 471–476.
21. S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and intelligent laboratory systems*, vol. 2, no. 1-3, pp. 37–52, 1987.
22. I. Jolliffe, *Principal component analysis*. Wiley Online Library, 2002.
23. S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *science*, vol. 290, no. 5500, pp. 2323–2326, 2000.
24. L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. Nov, pp. 2579–2605, 2008.
25. S. Lacoste-Julien, F. Sha, and M. I. Jordan, "Disclda: Discriminative learning for dimensionality reduction and classification," in *Advances in neural information processing systems*, 2009, pp. 897–904.
26. O. Thonnard, W. Mees, and M. Dacier, "Addressing the attack attribution problem using knowledge discovery and multi-criteria fuzzy decision-making," in *Proceedings of the ACM SIGKDD workshop on CyberSecurity and intelligence informatics*. ACM, 2009, pp. 11–21.
27. L. Van Der Maaten, "Fast optimization for t-sne," in *In Neural Information Processing Systems (NIPS) 2010 Workshop on Challenges in Data Visualization*, vol. 100, 2010.
28. Y. Bengio, J.-F. Paiement, P. Vincent, O. Delalleau, N. Le Roux, and M. Ouimet, "Out-of-sample extensions for lle, isomap, mds, eigenmaps, and spectral clustering," *Mij*, vol. 1, p. 2, 2003.
29. M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," in *NIPS*, vol. 14, no. 14, 2001, pp. 585–591.
30. J. B. Tenenbaum, V. De Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *science*, vol. 290, no. 5500, pp. 2319–2323, 2000.
31. M. A. Hearst, "Automatic acquisition of hyponyms from large text corpora," in *Proceedings of the 14th conference on Computational linguistics-Volume 2*. Association for Computational Linguistics, 1992, pp. 539–545.
32. P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *Journal of Machine Learning Research*, vol. 11, no. Dec, pp. 3371–3408, 2010.
33. G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.

34. M.-L. Zhang and Z.-H. Zhou, "Ml-knn: A lazy learning approach to multi-label learning," *Pattern recognition*, vol. 40, no. 7, pp. 2038–2048, 2007.
35. (2017, 01). [Online]. Available: http://www.ke.tu-darmstadt.de/resources/eurlex
36. E. L. Mencía and J. Fürnkranz, "Efficient multilabel classification algorithms for large-scale problems in the legal domain," in *Semantic Processing of Legal Texts*. Springer, 2010, pp. 192–215.
37. C. D. Manning, M. Surdeanu, J. Bauer, J. R. Finkel, S. Bethard, and D. McClosky, "The stanford corenlp natural language processing toolkit." in *ACL (System Demonstrations)*, 2014, pp. 55–60.
38. J. Seitner, C. Bizer, K. Eckert, S. Faralli, R. Meusel, H. Paulheim, and S. Ponzetto, "A large database of hypernymy relations extracted from the web," in *Proceedings of the 10th edition of the Language Resources and Evaluation Conference, Portoroz, Slovenia*, 2016.
39. G. Tsoumakas, I. Katakis, and I. Vlahavas, "Mining multi-label data," in *Data mining and knowledge discovery handbook*. Springer, 2009, pp. 667–685.