Parameterized Contrast in Second Order Soft Co-Occurrences: A Novel Text Representation Technique in Text Mining and Knowledge Extraction

Amir H. Razavi, Stan Matwin, Diana Inkpen, and Alexandre Kouznetsov
School of Information Technology and Engineering (SITE),
University of Ottawa, Ottawa, ON, Canada
{araza082, stan, diana, akouz086}@site.uottawa.ca

***Abstract*— In this article, we present a novel statistical representation method for knowledge extraction from a corpus containing short texts. Then we introduce the contrast parameter which could be adjusted for targeting different conceptual levels in text mining and knowledge extraction.**

The method is based on second order co-occurrence vectors whose efficiency for representing meaning has been established in many applications, especially for representing word senses in different contexts and for disambiguation purposes. We evaluate our method on two tasks: classification of textual description of dreams, and classification of medical abstracts for systematic reviews.

Keywords- Text Representation; Co-Occurrence Matrix; Text Mining; Concept Analysis; Knowledge Extraction

I. INTRODUCTION AND BACKGROUND

Machine knowledge extraction from text has been an attractive and applicable task since many years ago. We need a quantitative method to represent contexts in an expressive manner, in order to increase the performance of text mining and knowledge extraction. We use terms/words as the smallest meaningful unit of any context which plays a role in expressing meaning or intention through text. Therefore, capturing the right sense of any word in a context in the representation method is crucial. There are several hypotheses in the literature:

- You shall know a word by the company it keeps [27];
- Meanings of words are (largely) determined by their distributional patterns (Distributional Hypothesis [24])
- Words that occur in similar contexts will have similar meanings [17];

Most efforts on semantic extraction of words are focused on semantic similarity [25]: 'Automatically acquiring a relative measure of how similar a word is to known words […] is much easier than determining what the actual meaning is.' The Distributional Hypothesis [9, 10] says that words which occur in similar contexts tend to be similar.

In supervised text mining, the most common method for context representation is Bag-Of-Words (BOW). Texts are represented by the words they contain. If the absence or presence of a word is recorded, we call it a binary representation. If we use the frequency of the words, we call it a frequency representation. A normalized frequency representation is tf-idf [23].

In unsupervised concept learning and word sense disambiguation, in order to represent a given context, there are two approaches: first order co-occurrence vectors and second-order co-occurrence vectors.

In the first order context representation [4], we build a vector for any context containing a certain target word with an ambiguous sense. Any corpus word (feature) is represented by a position in the vector space. In each vector, we can see, whether any word in the corpus directly co-occurred with the target word (in that certain context) or not. Using the first order co-occurrence representation, by looking at the vectors, we can see which features directly contributed to the contexts in which the target word appeared.

There are two disadvantages of this method: first, very similar contexts may be represented by different dimensions in the feature space. Second, in short instances we will have too many zero features for machine learning supervised (classification) or unsupervised (clustering) task.

The second method for context representation, proposed by Schütze in 1998 [1], is called second order co-occurrence context representation. It is a more integrative method.

In our proposed method, we create a word-word co-occurrence matrix over the whole corpus (each row/column is a vector representation of the corresponding word) and then for representing any context we simply extract corresponding vectors for the words it contains. After averaging the vectors word by word, the average vector is called the second order co-occurrence vector of the context.

In the second-order co-occurrence, two terms that do not co-occur, will have some similarity if they co-occur with a third term. This is similar to the relation of a friend of a friend in social networks [17]. Synonyms are a special example of this kind of relationship. Although synonyms do not tend to occur in the same context (i.e. a short sentence), but they may occur in similar contexts and with the same neighboring words. This method helps confronting the data sparsity problem.

Although until now, the second order co-occurrence has been applied in variety of unsupervised purposes [4,5,6,7,8], for the first time we are going to apply a soft augmented version of it to a supervised text analysis task. We will specifically describe an implemented contrast parameter which can be helpful for representations in different tasks, with different targeted conceptual levels.

Experiments show that the second order context representation works better on limited volume of input data or localized scope [11], and the reason could be the high

sparsity of the first order representation which does not present enough discriminative information (due to many zero values for some dimensions in the vector space) for any recognition task. The second-order co-occurrence representation not only contains the main features (words/terms) of each context, but also contains many second order co-occurrence features. Therefore, the feature by feature co-occurrence matrix and consequently the context representation is less sparse than BOW and the first-order representation

When data is limited and sparse, exact features (as in the BOW method) in training and testing data, rarely occur in the same role. On the other hand, the second order co-occurrence captures and applies the indirect relations between features as well; therefore, it provides more information in order to increase the system's discriminative power.

Until now, this method has been applied mostly for unsupervised learning tasks like word sense disambiguation in a given context [12, 13, and 14] or short text/context clustering based on specified topics [15,16,].

II. METODOLOGY

We explain our Second Order Soft Co-Occurrence (SOSCO) method as an augmented implementation of the described method [1, 4] which is designed for short text corpus representation, (including more than one context in each entry) particularly for supervised text classification.

A. Preprocessing

In preprocessing, first all the headers, internet addresses, email addresses and tags has been filtered out and also all the extra delimiters like spaces, tabs, newline characters, and some characters like: "\ : ( ) ` 1 2 3 4 5 6 7 8 9 0 \ = \ [ ] / < > { } | ~ @ # $ \ % ^ & * _ + " have been removed from each text, whereas the expressive characters like: " - . , ; ' " " ' ! ? " were kept. Punctuation could be useful for determining the scope of speaker's speech. This step prevents us from including too many unrealistic tokens as features in the text representations.

B. Soft Co-Occurrence Matrix Creation

After preprocessing, we start tokenizing the corpus, in order to build a soft co-occurrence matrix in which the closeness of co-occurring pairs is recorded. The closeness is determined by considering a variety of configurations of any pair of words in a sentence (our window size). The configurations of the word pairs inside the sentences are:

1- Two adjacent words (bigrams regardless of their order)
2- Two consecutive words with one word in between.
3- Two consecutive words with more than one word in between.
4- Two consecutive words with coma "," interval in between.
5- Two consecutive words with semicolon ";" in between.
6- Two consecutive words with quotation " ' " or " " " in between.
7- Two consecutive words with "\r" or "\n" in between.

Note that we never have pairs of words with any of [. ! ?] in between, in a sentence.

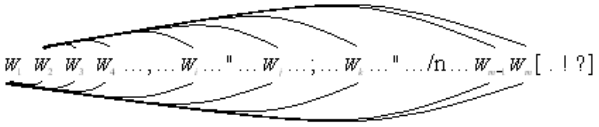

Figure 1. Illustrates the configurations of word pairs which can be extracted from a sentence.

Normally, co-occurrence is considered in a specific context or in a window of a limited size such as 3 to 7 words before or after a target word which indeed would restrict the total context size from 7 to 15 words.

We select sentences as our window size. On the other side, in order to minimize the noise interference on the matrix, we simply decrease the effect of a co-occurrence by increasing the above listed configuration number. In other words, except for the first configuration, the rest have a fraction of co-occurrence impact on the matrix.

If the number of tokens in a sentence is *n*, the number of pairs extracted from a sentence can be calculated as:

$$n+ (n-1)+ (n-2)+\ldots+1 = \frac{n(n+1)}{2} = \frac{1}{2}n^2 + \frac{1}{2}n$$

This shows that the computational complexity of building the soft co-occurrence matrix is of quadratic order of the typical sentence length which is less than 30 words $(30^2 = 900)$ and linear in the number of sentences in a corpus. We empirically observed the linear complexity, and it took a fraction of a second to process each short text.

Every individual condition listed above applies an assigned coefficient factor (weight) for accumulative closeness computation of any pair of words in the soft co-occurrence matrix. The corresponding weights for the above listed configurations are dramatically decreasing from top to bottom; the weights are assigned by adaptive learning.

There is an *Exclusive Or* relation between the configurations from bottom to top. This means that if condition #3 and #4 happen at the same time, we would only apply the smaller weight which corresponds to the larger one, which is #4.

The values in the matrix are calculated based on Dice's similarity measure for the closeness of a pair of words (X, Y) in a corpus, as follows:

$$c_{X,Y} = \frac{2\left(w_1 \cdot df_{1_{xy}} + w_2 \cdot df_{2_{xy}} + \ldots w_m \cdot df_{m_{xy}}\right)}{df_x + df_y}$$

$$df_x = df_{1_x} + df_{2_x} + \ldots + df_{m_x}$$

in which, C_{XY} is defined as the closeness of the pair of words (X,Y); w_i is the assigned weight for the

configuration number *i*; $df_{i_{xy}}$ is the frequency of co-occurrence of the pair (X,Y) in configuration *i* in the corpus; *m* is the number of distinct word pair configurations; df_x is the frequency of occurrence of the word X in the corpus; df_{i_x} is the frequency of occurrence of the word X in the configuration number *i* with any word in the corpus, and df_y is the frequency of occurrence of the word Y in the corpus and is calculated the same way as df_x. The values of C_{XY} in the matrix are not normalized at this stage; they will be normalized after building up the matrix.

Each row of the matrix is actually a descriptive vector that represents the closeness of the features that co-occurred with the particular word indicated by the row name.

C. General and Domain Specific Stop Words Removal

It is obvious that if we removed the stop words from the text prior to determine the configuration of each pair in it, we would have many changes among configuration numbers (one to three) and consecutively the corresponding effect to the co-occurrence matrix. In other words, with removing some stopwords as the first step some words which are actually located with one or more than one word in between could have been assigned a configuration that is adjacent or closer than reality, and in this way the algorithm will over estimate the degree of co-occurrence. In the implemented system, we just skip calculation when one of the pair members (or both) is in stoplist. We remove from the matrix the corresponding rows/columns in which all values are zeros, as we already skipped from those computations.

There are two groups of stopwords which indeed are removed: 1- general and 2- domain specific stop words.

First, we apply a general predefined stop word list appropriate for the domain we are working with (i.e., medical domain). Second, in some cases stop words are determined based on their frequency distribution, as detected from the corpus after generating of the word-word soft co-occurrence matrix. We remove words with very high frequency relative to the corpus size and term distribution in both classes and do not help to discriminate between them. We also remove words that appear only once in the corpus, as they will not help the classification, since they appeared only with one class, possibly by chance. As the word-word closeness is calculated regardless of words order in co-occurring pairs, the matrix is a symmetric matrix and the co-occurrences of any given word can be extracted from the corresponding row or column of the matrix, equally.

D. Two-Level Text Representation Vectors

In the first level, each *sentence* of a *short text* in the corpus is represented by averaging[1] the containing features' vectors, which are extracted from the soft co-occurrence matrix. In this matrix which is symmetric over feature space, for each word we can see all the words that co-occurred with it, over the corpus. At this step the soft co-occurrence matrix does not include stop words, hence the stop words cannot affect the creation of the representation vectors.

We employ the second order co-occurrence vectors (extracted from the above matrix) to perform an averaging process among the vectors of the words inside a sentence. The sentence representation vector at this stage has several times more non-zero features than the BOW representation of the same sentence. In the next level, we calculate the text representation vector by averaging the vectors of the sentences, which have been calculated during the first level. Performing this aggregation function (average) is another step toward increasing the number of non-zero elements of the text representation vector. Almost 90% of the features are non-zero by now. Although the value of any cell in the vector is an indicator of the association power of the corresponding feature in the vector space with the sentence/text that contains it, this value does not show directly if the feature occurred in sentence/text or not; it globally represents the relevance level of the sentence for each dimension (each feature).

E. Contrast Parameter

Browsing through a variety of text analysis projects, we see different conceptual levels targeted in each. Sometimes the task is to classify texts into topic classes such as medicine, agriculture, economy and so on. Other projects classify texts based on some restricted predefined conceptual domains or even based on sentiments or emotions that could be expressed by the writers. Obviously, for topic identification, there are some distinct keywords which play the essential role in the classification task, but in other cases such as sentiment or emotion analysis we cannot rely on these keywords.

Regarding the algorithm for creating our text representation vectors, implicitly we imposed smoothness among the feature space, versus the extreme contrast in the BOW representation of the same text. In a normalized BOW representation of a context we can see a non-zero value if a certain word explicitly occurs in the context and otherwise the value would be zero. However in the SOSCO representation of the same context, a value of a feature could be non-zero just if there is one (or more) co-occurred (not necessarily occurred) word (over the corpus) in that context and the value is directly related to the power of the closeness of the word occurred in the context and the feature in the feature space. Comparing the results of the two representations on several datasets/applications, we observed the advantages of each of them in different domains. Therefore we propose to define a contrast parameter that allows the value of each feature to vary between these two end points. Hence, if we define a range

[1] The averaging function can be changed with another aggregation function like maximum, upon the targeted conceptual level in the application and be substituted to the BOW in section [2.5].

of 0 up to 9 for the contrast parameter, we will have the BOW representation at the highest point of contrast (level 9) and the two levels (sentence and text) averaged second order soft co-occurrence vectors at the other end point of smoothness[2] (level 0 of contrast). We set the maximum value to 9 empirically, in order to have a limited number of values for searching the optimum value for each application.

We ran many experiments in a variety of applications on many input data with different targeted conceptual levels. We empirically observed that the optimum contrast value for the topic identification tasks is higher than the optimum one for some sentiment/emotion analysis tasks.

We also observed that applying different values for the contrast parameter may reveal about different aspects of the text which being represented. Hence the contrast parameter not only could be used for finding the most fitted representation of the text in any given application but also could be applied for obtaining a variety of representations of a given text for a committee of ensemble learners. (See part 5.2 for an example.)

F. An Example

If we want to illustrate the whole methodology with an example, we could start with a short text like (from the first dataset used in section 5.1):

"*It was Sunday. I was playing with the dog. All of a sudden my sister screamed and I fell down into the pool!*"

We build the sentence-based co-occurrence word-by-word matrix based on the described weights. Then, in a first step, we extract the corresponding vector for each word of the first sentence, out of the matrix which has one row for any word in the corpus. In second step, we calculate the average vector out of the three preliminary vectors for the three words in the first sentence. In this way, we obtain a representative vector for our first sentence.

We repeat the same process for the other two sentences of the text. We will have three individual average vectors at this stage. Finally, applying the proper contrast parameter value for the task we will have one text representation vector in which each value is between the two extreme ends of contrast and smoothness, and it is ready for participating in any learning process.

III. Method Specifications and Advantages

The basic co-occurrence method and its descendants have mostly targeted word sense disambiguation and topic detection tasks. Those generally were applied for unsupervised clustering tasks. Hence it is not easy to compare them with the current method, which is especially designed to be applied for supervised learning. We believe the following are the contributions specific to our method:

- Applying a proper value for the contrast parameter (in order to target different conceptual levels in different applications) can increase the discriminative power of any machine learning task based on this representation method.
- The capacity of the second order representation of a text includes more than the local context.
- It is robust in handling feature sparseness with only ~10% zero values for features *(the method represents texts based on co-occurrence of features in the whole corpus, rather a specified targeted word or topic).*
- Uses soft co-occurrence, applying different weights based on different word-word co-occurrence configurations.
- Increasing the representation power by building text representation vectors in two levels (sentence level and text level), instead of one level.
- Capacity of bypassing the LSI [26] dimension reduction procedure which usually is the most computational- and time- consuming step in similar tasks, because of using the fully loaded text vectors with less than 10% sparsity.
- The reduced feature space obtained after the above process is much more human-understandable than for LSI.
- The steps of SOSCO algorithm can be executed sequentially with linear complexity.

IV. Method Limitations

The described text representation method (SOSCO) is in contradiction with the independence assumption in some machine learning algorithms; therefore we will prefer to not use these algorithms (i.e., Naïve Bayes). Finding an appropriate contrast parameter value sometimes requires spending considerable time in the development step.

V. Experiments and Results

After testing the second order soft co-occurrence (SOSCO) representation on a variety of short texts corpora[3] and performing some preliminary modifications for improving the representation power, we applied the method on the following two text analysis tasks.

A. Classification of Emotional Tone of Dreams

Most of the studies on dreams have used time-consuming coding systems that depend on a rater's judgment. Hence, it is of interest to develop an efficient mean of scoring dreams that can be used with large data banks and reproduced across laboratories. The task of exploration of dream's emotional content using automatic analysis has been defined. A sample of 776 dreams, reported in writing by 274 individuals of varied age and sex, was used for word-correlation analysis.

A subset of 477 texts was rated by a judge using two 0–3 scales describing the negatively or the positive orientation of the dream.

A voting committee of different classifiers provided the most accurate results with the least mean squared error [19].

The agreement between machine rating and the human judge score on a scale of 0–3 was 64% (Mean Squared Error

[2] Note that both the BOW and second order co-occurrence representation vectors are already normalized (contain values between 0 and 1).

[3] The method has also been applied on some languages other than English.

0.3617), which represents 14% more than previous results on the same task, which was based only on the BOW representation method [21]. This was also significantly better than the chance probability of 25% and a baseline accuracy of 33%. The results indicate that estimates were at most one level away from human judge score[4] and offer a promising perspective for the automatic analysis of dream emotions, which is recognized as a primary dimension of dream construction.

B. Classifying Biomedical Abstracts Using Collective Ranking Technique

A systematic review is a structured process for reviewing literature on a specific topic with the goal of distilling a targeted subset of knowledge or data. Usually, the reviewed data includes titles and abstracts of biomedical research articles that could be relevant to the topic. The source data is extracted from biomedical literature databases such as MEDLINE [18] by running queries with keywords selected by domain experts. The queries are purposefully too broad, so that no relevant abstracts are missed. The output includes around $\sim 10^4$ articles. A systematic review can be seen as a text classification problem with two classes: a positive class containing articles relevant to the topic of review and a negative class for articles that are not relevant.

The selected approach is based on using committees of classification algorithms to rank instances based on their relevance to the topic of review. Experiments were performed on a systematic review data set provided by TrialStat Corporation [22]. The source data includes 23334 medical articles pre-selected for the review. While 19637 articles have title and abstract, 3697 articles have only the title. The data set has an imbalance rate (the ratio of positive class to the entire data set) of 8.94%.

A stratified repeated random sampling scheme was applied to validate the experimental results. The data was randomly split into a training set and a test set five times and the test set representation files have been built only based on the training set feature space. On each split, the training set included 7000 articles (~30%), while the test set included 16334 articles (~70%). The results from each split were then averaged.

We applied two data representation schemes to build document-term matrices: BOW and SOSCO representation. CHI2 feature selection was applied to exclude terms with low discriminative power. The ranking approach allows selecting abstracts that are classified as relevant or non-relevant with high level of prediction confidence (not less than the average prediction performance of human experts).

We needed to achieve a high level of recall and precision of the Positive class. Applying the optimum contrast parameter could not help us achieve the acceptable level of both recall and precision. For this reason, we decided to focus on two tails of certainty (the certainty of being Positive or Negative) in our classification. We observed that the highest-contrast BOW representation performed well on the part with the high certainty for the Positive class (700 abstracts), while the SOSCO representation with contrast parameter zero, has better performance on the part with the high certainty for the Negative class (8000 abstracts).

Therefore, the prediction zone consists of 8700 articles (700 top-zone articles and 8000 bottom-zone articles) that represent 37.3% of the whole corpus (53.3% of test set). At the same time, the gray zone includes 7634 articles (32.7% of the corpus, which is left for human experts to classify; this can save considerable time for them, since usually systematic reviews are done entirely manually). A committee of five classifiers was applied on the BOW and the SOSCO representation, individually, and then the results were combined through a voting scheme [20].

The results after voting are presented in Table 1. The table is a confusion matrix where only the prediction zone articles are taken in to account. Positive articles included in the top zone are true positives (TP), while positive articles included in the bottom zone are false negatives (FN). Negative articles in the top zone are false positives (FP), and negative articles in the bottom zone are true negatives (TN).

TABLE I. CONFUSION MATRIX ON THE PREDICTION ZONES APPLYING ENSEMBLE BOW AND SOSCO BY VOTING IN A COMMITTEE OF CLASSIFIERS.

Zone	Number of Abstracts	Correctly Classified	Incorrectly Classified
Top	700	590 (TP)	110 (FP)
Bottom	8000	7946 (TN)	54 (FN)

Table 2 presents the recall and the precision results for the Positive class (the class of interest), based on the prediction zone confusion matrix from Table 1. Table 2 also includes the average recall and precision results for human expert predictions (considered individually). This shows that our method achieves a significant workload reduction (37.3%), while maintaining the required performance level.

TABLE II. PERFORMANCE EVALUATION

Performance Measure	Machine Learning results on the prediction zone	Average human reviewer's results
Recall on the Positive Class	91.6%	90-95%
Precision on the Positive Class	84.3%	80-85%

We verified the performance of using ensemble method over data representation techniques. We ran classifiers committee on the SOSCO data representation; after that we ran classifiers committee on the BOW data representation, and finally, we ran the classifiers committee on both data representations together. The results are shown in Table 3 and 4. The number of misclassifications, both False

[4] Literature shows between 57- 80% agreement among the human judgment in this area and range.

Positives and False Negatives, is significantly less for the ensemble of data representation techniques, than for either of them used alone. In the tables we can see the performance of each representation at each tail of certainty.

TABLE III. FALSE POSITIVES WITH RESPECT TO DATA REPRESENTATION METHODS.

Split Number	SOSCO	BOW	Ensemble (SOSCO and BOW)
1	138	127	110
2	138	186	108
3	118	117	101
4	143	119	113
5	160	130	119
Average	139.4	135.8	110.2

TABLE IV. FALSE NEGATIVES WITH RESPECT TO DATA REPRESENTATION METHODS.

Split Number	SOSCO	BOW	Ensemble (SOSCO and BOW)
1	55	78	53
2	55	119	48
3	55	96	55
4	71	72	50
5	68	101	62
Average	60.8	93.2	53.6

Since the machine learning prediction performance is generally on the same level as the human prediction performance, using the described system will lead to significant workload reduction for the human experts involved in the systematic review process.

VI. CONCLUSION AND FUTURE WORK

In the future, we are planning to add one step of context detection in order to determine our window size dynamically and build the representation vectors based on its component contexts, as currently our window size is based on sentences. Our proposed SOSCO method, with a proper contrast parameter value, can be used in different levels of semantic, sentiment and conceptual analysis tasks.

REFERENCES

[1] H. Schütze. Automatic word sense discrimination. Computational Linguistics, 24(1):97–123. 1998.

[2] T. Van de Cruys. Exploring Three Way Contexts for Word Sense Discrimination. Denmark: Contextual Information in Semantic Space Models (CoSMo), 2007.

[3] F. Fukumoto and Y. Suzuki. Word sense disambiguation in untagged text based on term weight learning. In Procs. of the Ninth Conference of the European Chapter of the ACL, 209–216, Bergen. 1999.

[4] T. Pedersen and R. Bruce. Distinguishing word senses in untagged text. In Procs. of the 2nd Conf. on Empirical Methods in Natural Language Processing, 197–207, Providence, RI, August 1997.

[5] A. Purandare and T. Pedersen. Word sense discrimination by clustering contexts in vector and similarity spaces. In Procs. of Conf. on Computational Natural Language Learning, Boston, MA, 2004.

[6] A. Kulkarni: Unsupervised Discrimination and Labeling of Ambiguous Names. In Procs. of ACL 2005.

[7] T. Pedersen, A. K. Kulkarni, R. Angheluta, Z. Kozareva and Th. Solorio. An Unsupervised Language Independent Method of Name Discrimination Using Second Order Co-occurrence Features - Appears in the Procs. of CICLing 2006, Volume 3878 of LNCS, Springer, Mexico City, Mexico. February 19-25, 2006.

[8] A. Kulkarni and T. Pedersen. Name discrimination and email clustering using unsupervised clustering and labeling of similar contexts. In Procs. of the Second Indian International Conference on Artificial Intelligence, 703–722, Pune, India, December 2005.

[9] Z. Harris. Distributional structure. In Katz, J.J., ed.: The Philosophy of Linguistics. Oxford University Press 26–47, 1985.

[10] S. Mcdonald, M. Ramscar. Testing the distributional hypothesis: The influence of context on judgements of semantic similarity. In Procs. of the 23rd Annual Conf. of the Cognitive Science Society, 2001.

[11] A. K. Kulkarni. Unsupervised Discrimination and Labeling of Ambiguous Names. Proceedings of the ACL Student Research Workshop, 145–150, Ann Arbor, Michigan, June 2005.

[12] T. Pedersen and A. Kulkarni. Selecting the right number of senses based on clustering criterion functions. In Procs. of the Posters and Demo Program of the Eleventh Conf. of the European Chapter of the ACL, 111–114, Trento, Italy, April 2006.

[13] T. Pedersen and R. Bruce. Knowledge lean word sense disambiguation. In Proceedings of the Fifteenth National Conference on Artificial Intelligence, 800–805, Madison, WI, July. 1998.

[14] A. Purandare and T. Pedersen, SenseClusters - Finding Clusters that Represent Word Senses. In Procs. of AAAI-04, 1030-1031, San Jose, CA (Intelligent Systems Demonstration), July 25-29, 2004.

[15] T. Pedersen, A. Kulkarni, R. Angheluta, Z. Kozareva, and T. Solorio. An unsupervised language independent method of name discrimination using second order co-occurrence features. In Procs. of CICLing, 208–222, Mexico City, February 2006.

[16] T. Pedersen, A. Kulkarni,: Unsupervised discrimination of person names in web contexts. In Procs. of CICLing 2007, 299-310, 2007.

[17] G. Miller, and W. Charles, Contextual correlates of semantic similarity. Language and Cognitive Processes 6(1):1–28. 1991.

[18] MEDLINE. Available at: http://medline.cos.com

[19] A. H. Razavi, R. Amini, C. Sabourin, J. Sayyad Shirabad, D. Nadeau, S. Matwin & J. De Koninck - Evaluation and Time Course Representation of the Emotional Tone of Dreams Using Machine Learning and Automatic Text Analyses. In Proceedings of the 19th Congress of European Sleep Research Society, 2008.

[20] A. Kouznetsov, S. Matwin, D. Inkpen, A. H. Razavi, O. Frunza, M. Sehatkar and L. Seaward, "Classifying Biomedical Abstracts Using Committees of Classifiers and Collective Ranking Techniques", The 22th Canadian AI, May 2009.

[21] D. Nadeau, C. Sabourin, ,J. De Koninck, , S Matwin, , P. D. Turney, Automatic dream sentiment analysis, Proceedings of the Workshop on Computational Aesthetics at AAAI-06, Boston, USA 2006.

[22] TrialStat corporation web resources http://www.trialstat.com/

[23] K. Spark Jones, A statistical interpretation of term specificity and its application in retrieval. J. of Documentation 28 (1): 11–21, 1972.

[24] Z. Harris,: Distributional structure. In Katz, J.J., Fodor, J.A., eds.: The Philosophy of Linguistics, New York, Oxford University Press, 1964.

[25] C. D. Manning and H. Schütze. Foundations of Statistical Natural Language Processing. The MIT Press, Cambridge, MA. 1998.

[26] S. C. Deerwester, S. T. Dumais,., T. K. Landauer,., G. W. Furnas, R. A. Harshman, Indexing by latent semantic analysis. Journal of the American Society of Information Science 41 (6), 391- 407, 1990.

[27] J.R. Firth et al. Studies in Linguistic Analysis. A synopsis of linguistic theory, 1930-1955. Special volume of the Philological Society. Oxford: Blackwell, 1957.