Active Learning for Conversational Interfaces in Healthcare Applications

Aki Härmä[1(✉)], Andrey Polyakov[2], and Ekaterina Artemova[3]

[1] Philips Research, Eindhoven, The Netherlands
aki.harma@philips.com
[2] Philips Research, Moscow, Russia
[3] Sberbank, Moscow, Russia

Abstract. In automated health services based on text and voice interfaces, there is a need to be able to understand what the user is talking about, and what is the attitude of the user towards a subject. Typical machine learning methods for text analysis require a lot of annotated data for the training. This is often a problem in addressing specific and possibly very personal health care needs. In this paper, we propose an active learning algorithm for the training of a text classifier for a conversational therapy application in the area of health behavior change. A new active learning algorithm, Query by Embedded Committee (QBEC), is proposed in the paper. The methods are particularly suitable for the text classification task in a dynamic environment and give a good performance with realistic test data.

1 Introduction

The application context of the current paper is the development of automated therapeutic conversational interventions for behavior change [2], in particular, related to substance abuse. Counseling is known to be the most effective intervention for many lifestyle diseases, but counseling sessions are expensive for the health care system and often inconvenient for patients. Automation of the effective mechanisms of counseling by automated agents would lead to better coverage and cost savings. In a typical application, a conversational agent would implement some elements of the Cognitive Behavioral Therapy [9]. Typically, the agent would be available through a social media platform possibly with a speech interface. The text understanding system should be able to detect the topics and sentiment structures relevant to the control of the conversation according to the selected therapeutic strategy.

Recurrent neural networks are popular for text understanding, but they require a large corpus of labeled training data, which is difficult to collect. Also, natural language communication is an example of a non-stationary learning environment where the evolution in the conversational culture over time and populations require local customization and maintenance of the classifier, possibly even at the level of an individual customer. The client talk related to a particular substance may be very specific and patients may even develop a personal

F. Koch et al. (Eds.): AIH 2018, LNAI 11326, pp. 48–58, 2019.
https://doi.org/10.1007/978-3-030-12738-1_4

vocabulary to discuss about the addiction. Also, the content is naturally very sensitive and it is therefore desired that the client talk should not be uploaded to a cloud processing platform but processed locally in the client device.

One approach for the maintenance and continuous improvement of a classifier in the production environment is to use *active learning* (AL) methods [3,16]. In pool-based AL methods, only a small part of the available content is manually labeled and used to train the classifier. A typical approach is to use a committee of classifiers [12] to select items that are *difficult to classify* based on the current statistics. This approach works well in many conventional problems but often leads to robustness problems that are common in many deep learning architectures [6]. Also, while the classification of client talk may be, at least in the future Edge AI technologies, performed the client device, the detection of novel training content selection based in AL in a client device is significantly more challenging.

In this paper, we demonstrate an application of active learning in the classification of short text messages, *tweets*, from a social media platform using a text classifier based on Recurrent Neural Networks, RNNs [8]. We propose an algorithm for the pool-based selection where the committee method is applied in a latent variable space. In particular, the committee is embedded in a space spanned by the class likelihoods of the last classifier. In this paper, the method is called Query by Embedded Committee, QBEC. The method is computationally significantly lighter than the conventional Query-by-Committee, QBC, method.

2 Sample Selection Methods

In pool-based active learning [16] new samples are selected to the training data from a large pool of unlabeled content. The selection may be based on different principles and aim at selecting the most informative or representative samples [10,17], reduce the variance of the classification errors [3], or *diameter* in a space spanned by alternative classifiers [4].

The AL process starts with an initial set P_0 of labeled tuples of K feature vectors x_k and corresponding labels l_k, i.e.,

$$P_0 = \{x_k, l_k\}, k = 0, \cdots, K-1 \tag{1}$$

The Initial classification model M_0 is developed using P_0. Next, a new set S_1 is selected from the pool. The samples are manually labeled by a human oracle, for example, a health counselor. The new training data P_1 is produced by adding the samples S_1 to P_0. The model is updated and deployed. The same update cycle can then be continuously repeated.

The selection of the next batch S_{j+1} of B samples can be based on many different criteria. The minimum requirements for a j^{th} iteration are

1. *novelty:* $P_j \cap S_{j+1} = \emptyset$
2. *richness:* $x_n \neq x_m, \forall n, m \in S_{j+1}$

i.e., the B new samples in S_{j+1} should be novel and they should be different from each other.

2.1 Query by Committee

In the popular Query-by-Committee (QBC) method [12] the novelty condition is addressed by measuring the disagreement in a committee of R different classifiers C_r trained using P_j.

$$d_j = \mathcal{D}\left[C_0(x_k), C_1(x_k), \cdots, C_R(x_k)\right] \tag{2}$$

where $\mathcal{D}[]$ is some measure to compute the disagreement.

In a typical case, the disagreement is based on vectors of class likelihoods given by the classifiers $\mathbf{p}_r(x_k) = C_r(x_k)$. In a committee of two classifiers, the disagreement can be defined as a norm of the difference $d_k = |\mathbf{p}_0(x_k) - \mathbf{p}_1(x_k)|$.

Algorithm 1. Query by Committee

Require: Label the initial data set P_j and set $j = 0$
1: **repeat**
2: Train the main classifier model M_j using P_j data
3: Train the committee classifiers C_0, C_1 using P_j data
4: Compute the disagreement $d_k \in S$ in the new batch
5: Pick K samples from S which has the highest disagreement using the knockout neighborhood penalization described above.
6: Labeling of the new samples by a human expert
7: Add new samples to the training data $P_{j+1}, j = j + 1$
8: **until** Stopping criteria are met.

The committee often disagrees on very similar samples, and therefore the basic algorithm does not provide the required *richness* for the new sample collection. A *pareto* optimal solution is needed to meet both the *novelty* and *richness* conditions. The richness is related to the nearest neighbor problem (NN). The k-nearest-neighbor searching problem (kNN) is to find the k -nearest points in a dataset $X \subset \mathbb{R}^d$ containing n points to a query point $q \in \mathbb{R}^d$ under some norm. There are several effective methods for this problem when the dimension d is small (e.g. 1, 2, 3), such as Voronoi diagrams [18] or Delaunay triangulation [5]. When the dimension is moderate (e.g., up to the 10's), it is possible to use *kd* trees [8] and metric trees [11]. If the dimension is high, then Locality-Sensitive Hashing (LSH) is the very popular method used in applications. In the current paper, we use an iterative algorithm where the new samples that are close to already selected samples are penalized. Experiments with other sampling principles is a part of future work.

2.2 Query by Embedded Committee

The selection of the new samples based on a disagreement of a committee assumes a certain variability among the committee members [12]. This is typically achieved by using different initialization of the classifiers C_j, or by using

different classifier prototypes or kernels. In the case of a complex model, for example, based on multiple layers of memory networks and dense layers, the training of a committee can be a large effort and may take, for example, several hours of processing time in a GPU. In principle, the training of each committee member model takes as much time as the training of the main model itself. However, the final scoring of the network is light and can be performed, for example, in a smartphone or another end-user device.

In this paper, the proposed method is to use the committee in another feature space derived from the outputs of the model. A multi-class classifier is often developed using the one-hot encoding principle where the classifier produces a vector of class likelihoods $\mathbf{p}_r(x_k) = M(x_k)$ for a feature vector x_k. The likelihoods represent the class predictions for the testing data. In a geometric sense, the likelihood vectors $\mathbf{p}_r(x_k)$ span an orthonormal space, a *class space* of the current classifier, where each axis represents a class.

The proposed method is a variation of the QBC method where the selection task is performed in the class space of the current classifier. The class space is a metric low-dimensional space, and there the committee can be based on conventional classification tools, e.g., based on a random forest or another relatively light algorithm. The training of the committee of classifiers and testing of them on a new data can be performed in an end-user device. Therefore, this enables local active learning of the classification model.

The processing steps of the proposed method are described in Algorithm 2 below.

Algorithm 2. Query by Embedded Committee (QBCSC)

1: **repeat**
2: Use classifier M_j to get class likelihood vectors for data $\mathbf{p}_r(x_k) \forall x_k \in P_j$
3: Use QBC method defined in the class space to select the new samples for labeling.
4: Add new samples to the training data $P_{j+1}, j = j + 1$
5: **until** Stopping criteria are met.

In this paper we call the modified method Query by Embedded Committee (QBEC), to separate it from the conventional QBC. In the current paper, the committee is *embedded* in the class space. Naturally the same can also be performed in another output space, for example, corresponding to intermediate layers of the network.

2.3 Computational Load

QBEC method works faster then QBC due to hypothesis space reduction. Namely, it has been shown in [7] that the number of queries for labels that the algorithm will make is $\mathcal{O}(\frac{d}{g} log(\frac{1}{\varepsilon}))$, where d is the Vapnik-Chevonenkis dimension, g is some constant, ε is required accuracy.

If QBC works in $\mathbb{R}^n$, then $d = n + 1$ (as the hypothesis space is divided by set of oriented hyperplanes). Simultaneously, QBEC works in $\mathbb{R}^k$, where $k \ll n$

is the dimension of a space which is spanned by the class likelihoods of the current classifier. So, the corresponding Vapnik-Chervonenkis dimension will be $m = k + 1$. So, the complexity of QBEC is $\mathcal{O}(\frac{m}{g} log(\frac{1}{\varepsilon})) < \mathcal{O}(\frac{d}{g} log(\frac{1}{\varepsilon}))$.

3 Experiments

Let us start with a synthetic example to illustrate the differences between QBC and the proposed QBEC method, and their benefits over random sampling from the pool.

3.1 Synthetic Example

The original synthetic data is shown in Fig. 1(a) with two classes illustrated by red crosses and blue circles. A random forest (RF) classifier was designed for the set P_0 with 100 labeled samples. In the QBC method a committee of two RF classifiers was designed using a different initialization. The selection of new samples was based on selection of the samples with the largest difference in the class likelihood values between the classifiers. Figure 1(b) shows an example of a

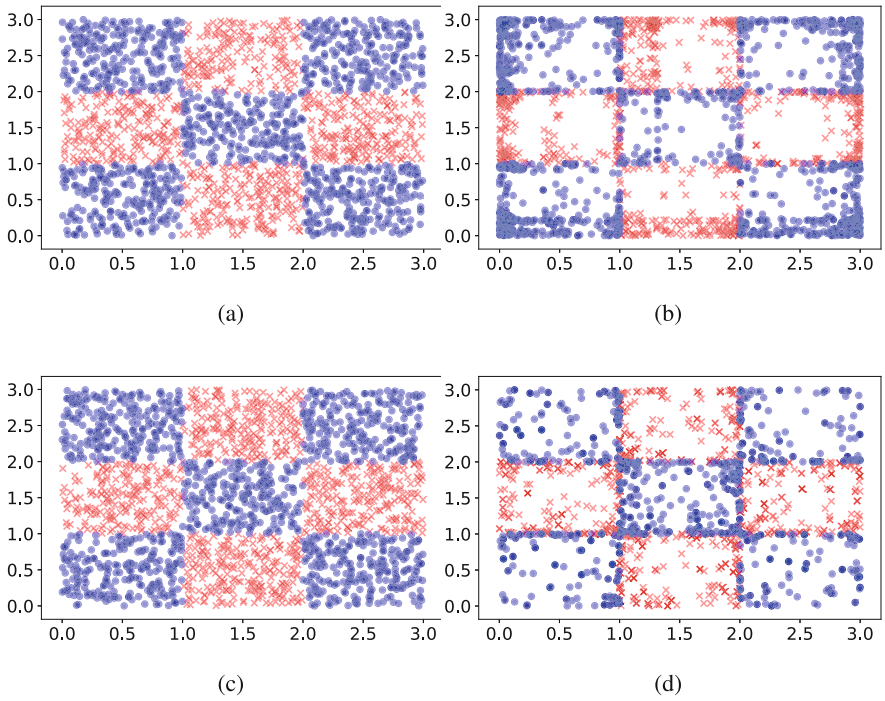

Fig. 1. (a) test data, (b) QBC samples, (c) random samples, (d) QBEC sampling. The x and y axes are arbitrary coordinates. (Color figure online)

QBC sampling. A random sampling used in the reference condition is illustrated in Fig. 1(c). The QBC method clearly takes more samples from the class borders that the random sampling method. The QBEC method also focuses on the class borders but puts more emphasis on the borders between classes rather than outer borders.

The accuracy in the training in the three methods is shown in Figs. 2(a) and 2(b). The QBC and QBEC have a similar performance in the first batches, but the accuracy of QBEC method keeps improving at the point where the performance of QBC saturates. This may be understood in this case by comparing the selections in the two methods in Fig. 1(b). In the QBCSC the sampling focuses on borders between the classes while in the conventional QBC solution a large number of samples are selected from the outskirts of the feature space which is less relevant for the class confusions measured by the accuracy.

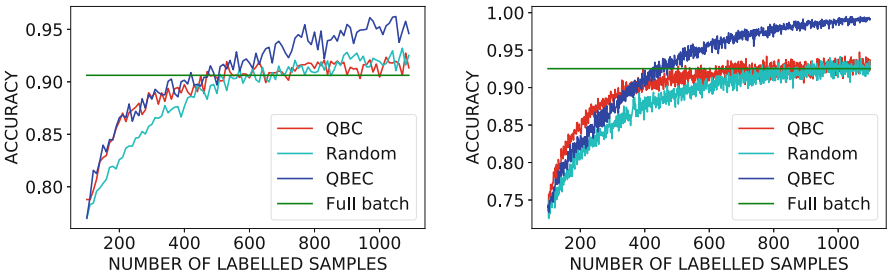

Fig. 2. Accuracy in an iterative active learning experiment using the three methods. The straight green line corresponds to the accuracy of a single classifier trained with the full set of 600 labeled samples. Batch sizes are (a) 1 (b) 10 (Color figure online)

4 Experiment with Tweets

In this paper, the content is from *Twitter*, which is a popular short-text messaging platform. The content was selected by keywords that relate to smoking and tobacco use. In the typical flow of content, the test system gave approximately 1000 tweets per day when excluding repetitions (*re-tweets*) of the same message (Fig. 3).

In the current paper the content is manually classified into three classes: *sustain talk*, *change talk*, and *neutral* communication. The two first classes are considered important elements in many therapeutic techniques for substance abuse, such as CBT [9] or Motivational Interviewing (MI) [14]. The target behavior is to reduce or quit smoking. Sustain and change talk contains all client talk that speaks against or for the target behavior, respectively. The neutral class contains all other content with the same keywords. The data contains all English language

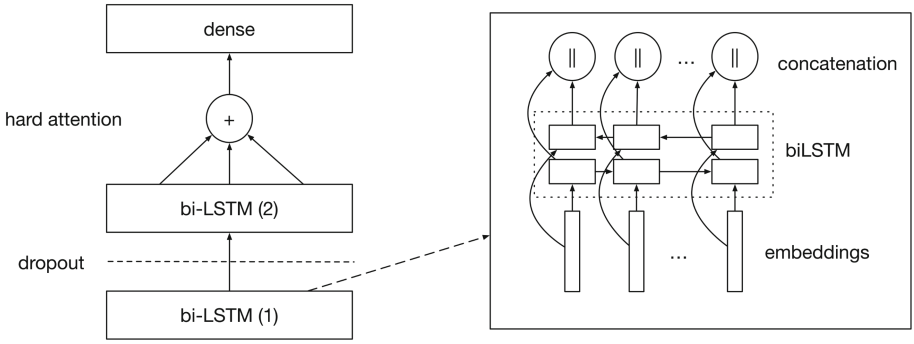

Fig. 3. Architecture of the deep learning neural network used for text classification. **On the left**: the general architecture of the deep learning neural network. **On the right**: the input layer and the first Bi-LSTM Layer. The second Bi-LSTM copies the first one, except the input are not embeddings, but the output of the first Bi-LSTM layer. See Table 1 for the number of units and other parameters.

messages from Oct. 2017 until the end of Jan 2018[1]. There are cultural elements in the data. For example, the tweets from October contain messages that relate to the *Stoptober* smoking cessation campaign in the UK and other countries, in December there are tweets from people who plan to *quit for January*, and there are several referrals to a popular song called *cigarette daydreams*.

4.1 Text Classification System

In this paper, we use a typical architecture for a text classifier based on a state-of-the-art deep learning RNN tools. The text classifier model has six components, presented in Table 1.

Table 1. Architecture of the deep learning neural network used for text classification

Layer	Parameters
Input: embedding layer	Google SGNS [13], Stanford GloVe [15]
Bi-LSTM	64 LSTM units
Dropout	0.2
Bi-LSTM	32 LSTM units
Hard attention layer [1]	
Output: dense layer	Regular dense layer with softmax activation

[1] The ethical and legal approval of the data collection was granted, and handled according to, by the Internal Committee for Biomedical Experiments (ICBE) of Philips.

The embedding layer is meant to map each word of the input text into a low dimensional embedding vector, while the bidirectional layers get higher level features from the input, dropout being used for regularization. The hard attention layer is used for global re-weighting of hidden layers, and the desired class label is chosen using a regular dense layer with softmax activation.

5 Results

The initial classifier $C^{(0)}$ was trained using a manually classified set of 2398 tweets. Examples of typical tweet types and their counts in the initial training set are shown in Table 2. Also, an independent test data set with manually labeled tweets was used for testing. The performance of $C^{(0)}$ in an independent training set is poor; the accuracy is barely above 0.5. In the following experiment, the active learning process was executed sequentially so that the current dump of tweets about the target topic was downloaded once a day, classified using classifier $C^{(n)}$. Approximately one percent, typically around 30 tweets, were selected to the manual labeling using one of the selection methods. The samples were manually labeled and included in the training set, and subsequently used in the training of the next model $C^{(n+1)}$.

Table 2. Examples of tweets and their counts in the initial training set.

Talk type	Example	#samples
Change talk	Two weeks without smoking!	246
Sustain talk	I'm having a cigarette	514
Neutral	A man was smoking outside	1651

The latent space representation formed by the outputs of likelihoods at the output layer of the network is illustrated in Fig. 4. The three talk types are separated in the latent variable space.

The numbers of new labeled samples resulting from daily 23 iterations in the three methods are illustrated in Figs. 5a–c. First, it seems that random selection rarely picks samples from change talk category while those are much more common in the two other methods. The accuracy in the three methods, respectively, is shown in Fig. 5d. In the random selection the accuracy does not improve over the iterations, but in the two other methods, there is a clear improving trend. Unlike the results with the checker board data, there is not really a difference in accuracy between QBC and QBEC, although, the computational requirements and processing time in QBEC is obviously significantly lower than in QBC.

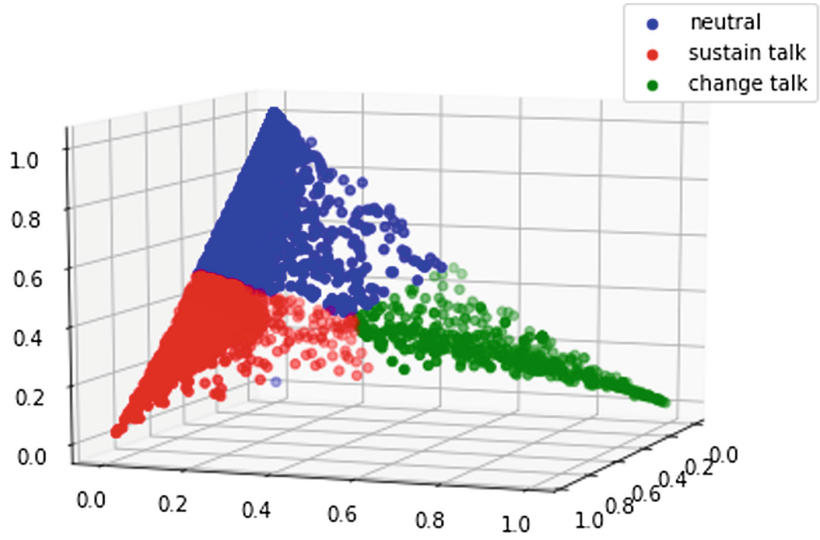

Fig. 4. Example of three classes of tweets in the 3D latent variable space.

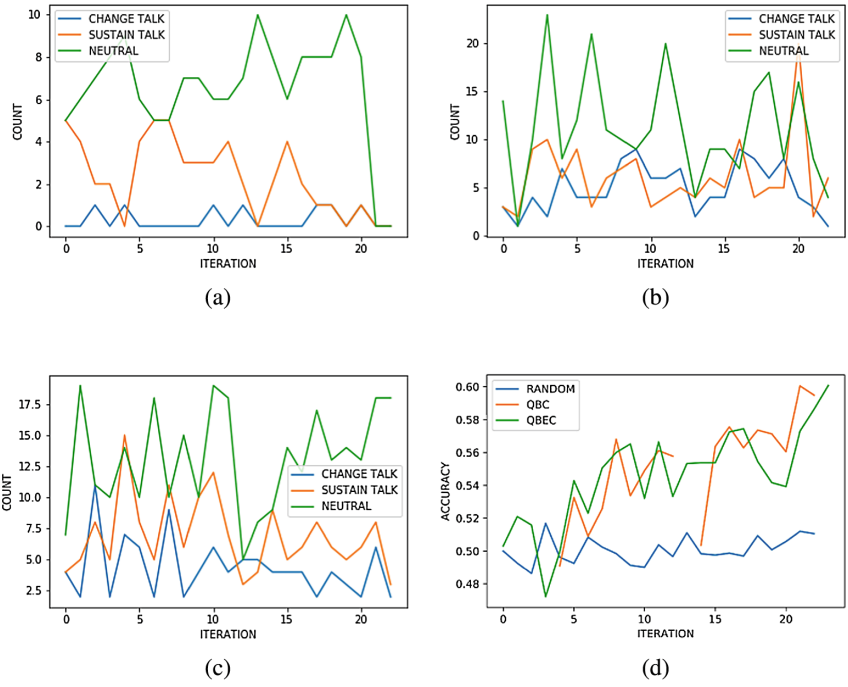

Fig. 5. (a) Random selection (b) QBC, and (c) QBEC, (d) accuracy in the three methods based on correct detections. X-axis represents the iteration number (days of downloaded tweets). One data point in QBC curve is missing due to an error in data handling.

6 Conclusions

In this paper, we propose a new algorithm for active learning in the application of text classification in the application of health counseling. The training of a classifier for a specific complex talk type requires a large labeled database which is typically difficult, expensive, and time-consuming. There may also continuous concept drift in the target area, for example, due to various cultural influences.

A popular approach for active learning is to use a disagreement in a committee of classifiers to select samples for manual labeling and inclusion into the training. These methods are commonly called Query-by-Committee (QBC) methods. The QBC methods require that multiple classifiers are trained for the task. In applications where the classifier is complex, a.e.g, a deep neural network model, and requires a long training time this may be problematic. In the method introduced in the current paper, the committee selection is performed in a low dimensional space spanned by the likelihoods of the current classifier model. In this case, the actual classifiers of the committee can be fairly simple. The method is called Query-by-Embedded-Committee (QBEC).

We demonstrate the performance of QBEC first using synthetic data. The performance of QBEC turns out to be superior to the random selection of training samples and it, surprisingly, exceeds the performance of QBC. One may speculate that this is because the embedding based on the prediction likelihoods inherently zooms the committee to *zoom* into areas where the disagreement is largest.

In a second experiment, we trained a complex classifier for classification of tweets related to smoking behavior into three classes. The classes represent change talk, sustain talk, and neutral communication of the talker about tobacco use. This is a very challenging classification problem requiring a large labeled database. In the active learning experiment, 1% of tweet content downloaded on each day was manually labeled and included in the new model. It was shown that QBC and QBEC outperform random selection of samples. However, the results of the two methods are similar. However, it should be noted that the computational of QBEC is significantly lower than in QBC. Therefore, the sample selection in QBEC could be performed even in a customer device such as a smartphone.

References

1. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473 (2014)
2. Bickmore, T., Giorgino, T.: Health dialog systems for patients and consumers. J. Biomed. Inform. **39**(5), 556–571 (2006). https://doi.org/10.1016/j.jbi.2005.12.004
3. Cohn, D.A., Ghahramani, Z., Jordan, M.I.: Active learning with statistical models. J. Artif. Int. Res. **4**(1), 129–145 (1996). http://dl.acm.org/citation.cfm?id=1622737.1622744
4. Dasgupta, S.: Coarse sample complexity bounds for active learning. In: NIPS 2005, pp. 235–242. MIT Press, Cambridge (2005). http://dl.acm.org/citation.cfm?id=2976248.2976278

5. Delaunay, B.: Sur la sphere vide. a la memoire de georges voronoi. Bulletin del'Academie des Sciences del'URSS (6), 793–800 (1934)
6. Fawzi, A., Moosavi-Dezfooli, S.M., Frossard, P.: The robustness of deep networks: a geometrical perspective. IEEE Sig. Process. Mag. **34**(6), 50–62 (2017). https://doi.org/10.1109/MSP.2017.2740965
7. Freund, Y., Seung, H.S., Shamir, E., Tishby, N.: Selective sampling using the query by committee algorithm. Mach. Learn. **12**(28), 133–168 (1997)
8. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
9. Hofmann, S.G., Asnaani, A., Vonk, I.J., Sawyer, A.T., Fang, A.: The efficacy of cognitive behavioral therapy: a review of meta-analyses. Cogn. Ther. Res. **36**(5), 427–440 (2012). https://doi.org/10.1007/s10608-012-9476-1, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3584580/
10. Hoi, S.C.H., Jin, R., Lyu, M.R.: Batch mode active learning with applications to text categorization and image retrieval. IEEE Trans. Knowl. Data Eng. **21**(9), 1233–1248 (2009). https://doi.org/10.1109/TKDE.2009.60
11. Liu, T., Moore, A.W., Gray, A., Yang, K.: An investigation of practical approximate nearest neighbor algorithms. In: Saul, L.K., Weiss, Y., Bottou, L. (eds.) Advances in Neural Information Processing Systems, pp. 825–832. MIT Press, Cambridge (2015)
12. McCallum, A., Nigam, K.: Employing EM and pool-based active learning for text classification. In: ICML 1998, pp. 350–358. Morgan Kaufmann Publishers Inc., San Francisco (1998). http://dl.acm.org/citation.cfm?id=645527.757765
13. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
14. Miller, W.R., Rose, G.S.: Toward a theory of motivational interviewing. Am. Psychol. **64**(6), 527–537 (2009). https://doi.org/10.1037/a0016830, http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2759607/
15. Pennington, J., Socher, R., Manning, C.: Glove: global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543 (2014)
16. Tong, S., Koller, D.: Support vector machine active learning with applications to text classification. J. Mach. Learn. Res. **2**, 45–66 (2002). https://doi.org/10.1162/153244302760185243
17. Wang, L., Hu, X., Yuan, B., Lu, J.: Active learning via query synthesis and nearest neighbour search. Neurocomputing **147**(Suppl. C), 426–434 (2015). https://doi.org/10.1016/j.neucom.2014.06.042, http://www.sciencedirect.com/science/article/pii/S0925231214008145
18. Ying, S., Xu, G., Li, C., Mao, Z.: Point cluster analysis using a 3D Voronoi diagram with applications in point cloud segmentation. Int. J. Geo-Inf. (ISPRS) **4**(3), 1480–1499 (2015)