2016

Advanced analytics for the automation of medical systematic reviews

Prem Timsina
Dakota State University

Jun Liu
Dakota State University

Omar F. El-Gayar
Dakota State University

Recommended Citation

Timsina, P., Liu, J., & El-Gayar, O. (2016). Advanced analytics for the automation of medical systematic reviews. Information Systems Frontiers, 18(2), 237-252.

Inf Syst Front (2016) 18:237–252
DOI 10.1007/s10796-015-9589-7

Advanced analytics for the automation of medical systematic reviews

Prem Timsina[1] · **Jun Liu**[1] · **Omar El-Gayar**[1]

Published online: 18 August 2015

Abstract While systematic reviews (SRs) are positioned as an essential element of modern evidence-based medical practice, the creation and update of these reviews is resource intensive. In this research, we propose to leverage advanced analytics techniques for automatically classifying articles for inclusion and exclusion for systematic reviews. Specifically, we used soft-margin polynomial Support Vector Machine (SVM) as a classifier, exploited Unified Medical Language Systems (UMLS) for medical terms extraction, and examined various techniques to resolve the class imbalance issue. Through an empirical study, we demonstrated that soft-margin polynomial SVM achieves better classification performance than the existing algorithms used in current research, and the performance of the classifier can be further improved by using UMLS to identify medical terms in articles and applying resampling methods to resolve the class imbalance issue.

Keywords Healthcare · Medical systematic reviews, analytics · Support vector machines

1 Introduction

Evidence Based Medicine (EBM) refers to the application of state-of-the-art medical evidence to improve the quality and reduce the cost of medical care (Cohen et al. 2010). Although the classical vision of EBM required physicians to directly search the relevant medical research for evidence applicable to their patients, the modern conception of EBM heavily relies on synthesis of research findings in the form of an evidence report commonly referred to as a systematic review (SR). According to Higgins and Green (2011), "a systematic review is a high-level overview of primary research on a particular research question that tries to identify, select, synthesize and appraise all high quality research evidence relevant to that question in order to answer it". Each systematic review addresses a clearly formulated problem. As an example, (Couch et al. 2008) presents a systematic review of "diabetes education for children with Type 1 Diabetes Mellitus and their families". It synthesizes the findings presented in 80 pertinent articles. Nowadays, systematic reviews form a key resource for informing evidence based medical practice. With the increasingly rapid pace by which medical knowledge is created, researchers, practitioners and policy makers are challenged to keep pace with state-of-the-art medical evidence and incorporate such evidence into practice. Systematic reviews respond to this issue by recognizing, appraising, and synthesizing research-based evidence from multiple sources and presenting it in an accessible format (Mulrow 1994).

Developing a medical systematic review is a much more demanding, rigorous, and resource-intensive process than develop a literature review in other domains, since systematic reviews attempt to bring a high level of rigor to reviewing research evidence and are often developed based on a peer-reviewed protocol so that they can be replicated if necessary. Surprisingly, the current workflow for creating and updating SRs is largely a manual process. An initial search by querying databases such as Medline, Cochrane and Embase often returns a large number of articles given a medical topic. Developing the review presented in (Couch et al. 2008) first

✉ Prem Timsina
ptimsina@pluto.dsu.edu

Jun Liu
jun.liu@dsu.edu

Omar El-Gayar
Omar.El-Gayar@dsu.edu

[1] College of Business and Information Systems, Dakota State University, 820 N. Washington Avenue, Madison, SD 57042, USA

involves retrieving 12,740 articles based on keywords such as diabetes, diabetic children, diabetic family members, and diabetes education in order to ensure that none of the relevant articles will be missed. These 12,740 articles were then evaluated manually by a team of scientists using highly methodic procedures. Only 80 of them were selected according to the inclusion and exclusion guidelines. Finally, the scientists synthesized the research findings in the 80 articles to establish the best education for children with Type 1 diabetes mellitus and their families. The articles that need to be included in a systematic review are usually selected in two steps. The first step is called abstract triage, where scientists identify "relevant" articles that can potentially be included in a SR based on the title and abstract of the articles. This phase of screening articles usually requires a long time and significant effort as it involves a group of scientists evaluating thousands of articles in order to find the relevant ones. The second step is referred to as full-text triage. It involves full text inspection of the relevant articles selected in the title/abstract triage to determine those that satisfy the inclusion criteria and will be included in a systematic review (Shojania et al. 2007). Due to the manual workflow of selecting articles for systematic reviews (SRs), developing SRs requires a significant investment in time (1139 expert hours on average) and funds (up to a quarter of a million dollars) from a dedicated and qualified research team (Allen and Olkin 1999; McGowan and Sampson 2005).

Nowadays, medical knowledge base is growing at an astounding pace. Reports of new clinical trials are being published at the rate of over 20,000 per year (Shojania et al. 2007). This creates an enormous challenge for scientists trying to develop and update systematic reviews to keep pace with the development in the medical field. Cochrane Collaboration estimates that at least 10,000 new systematic reviews are needed to cover most of the healthcare problems (Higgins and Green 2011). Unfortunately, fewer than half of this number has been published even after ten years of focused effort by the EBM community (Higgins and Green 2011). Once a review is created, the job is not done yet - a systematic review needs to be updated periodically (Cochrane 2013). The median time for a review to become obsolete is 5.7 years; for some medical conditions like cardiovascular, a SR may be obsolete in less than a year (Shojania. et al. 2007). A report published by Agency for Healthcare Research Quality (AHRQ) indicates that only 2 % of systematic reviews published in all journals represent updates of previous reviews (whether conducted by the same authors or not) (Shojania. et al. 2007). Researchers have attributed the difficulty of developing and updating systematic reviews to keep up with medical research advances to the aforementioned resource intensive manual process needed to screen articles (Shemilt et al. 2013). We lack highly refined automated tools that help reviewers sort and prioritize articles, which has been become a bottleneck that has hitherto constrained the timely creation and update of systematic reviews.

There are efforts that have leveraged text analytics (Bekhuis and Demner-Fushman 2012; Shemilt et al. 2013; Adeva et al. 2014) to automate the article screening procedure for systematic reviews. Most existing literature focuses on addressing a text classification problem, where medical articles are classified as relevant or irrelevant to the topic based on the title and abstract of the articles. As in any text classification task, we need to enhance both recall (i.e., among the articles that are deemed relevant and included in a systematic review, the fraction of those classified as "relevant") and precision (i.e., among the articles that are classified as "relevant", the fraction of those will actually be included in a review). Any automated system for identifying relevant articles must maintain a very high level of recall since a systematic review should include most, if not all, articles that provide high quality evidence relevant to the topic. Any system with a low recall would be of little use (Matwin et al. 2010). Precision is also essential in this context since a higher precision means that the articles that are classified as relevant are indeed relevant, which means that a smaller number of articles needed to be reviewed during the downstream full-text triage stage. Hence, in order to resolve the aforementioned bottleneck in the screening of medical articles, it is necessary to improve precision while maintaining a high recall. Among the existing research, a few studies such as (Bekhuis and Demner-Fushman 2012; Cohen et al. 2006; Matwin et al. 2010) attempted to achieve a high recall. Nonetheless, the results of these studies have shown a tendency for precision to decline as recall increases. Another conspicuous issue that has been largely ignored in existing research is that systematic review datasets are normally highly imbalanced, which means that among the thousands of articles to be selected, only a small number of them will be included in the final systematic review. The imbalance ratio ranges from 1:10 to 1:1000 (Shemilt et al. 2013). Class imbalances have been reported to hinder the performance of classifiers proposed in existing research. (Bekhuis and Demner-Fushman 2012; Cohen et al. 2006; Matwin et al. 2010).

The objective of this research is to develop an advanced analytics-based approach to automatically identifying relevant articles that could be included in systematic reviews based on the title and abstract of the articles. The proposed approach is primarily intended for updating exiting systematic reviews when a training dataset is readily available. It can also be used for systematic review creation if researchers can create a training dataset by manually reviewing a certain number of articles. Our text analytics based approach aims to improve the precision of article classification for systematic reviews while sustaining a very high level of recall. It makes three improvements to the existing methods described in literature. First, we propose to use the Unified Medical Language Systems

(UMLS) to extract medical terms as features for article classification, while the majority of existing research uses the "bag-of-words" approach (Adeva et al. 2014; Shemilt et al. 2013; Bekhuis and Demner-Fushman 2012; Wallace et al. 2010; Cohen et al. 2006). Our study demonstrated that the automatically extracted Unified Medical Language System (UMLS) terms helped boost classification performance. Second, we propose to use soft-margin polynomial Support Vector Machine (SVM) to classify articles. Using different medical datasets, we showed that soft-margin polynomial SVM achieved higher precision and recall, compared with several algorithms proposed in existing research. Third, to deal with the aforementioned class imbalance problem, we examined various re-sampling methods to re-sample the training data. The results of our comparative experiments indicate that a soft-margin polynomial SVM classifier that leverages more precise feature representation using UMLS and integrates the Synthetic Minority Oversampling (SMOTE) method (Chawla 2010) has the potential to yield significantly improved performance in identifying relevant articles for systematic reviews.

The remainder of the paper is organized as follows. The next section summarizes related work, followed by a discussion of the research gap we intend to address, a description of our research methodology, and a presentation and discussion of our experimental results. The last section concludes the article.

2 Related work

There have been some attempts in literature to leverage analytics to automate systematic reviewer generation and update (Ananiadou et al. 2009; Bekhuis and Demner-Fushman 2012; Cohen et al.; Frunza et al. 2010; Shemilt et al. 2013). One of the most significant research done in this area is one conducted by Cohen et al. (2006). In this National Institute of Health (NIH) supported project, Cohen et al. used the perceptron algorithm to identify journal articles for inclusion in systematic reviews based on the title and abstract of the articles. While the perceptron-based classifier achieved high recall, precision was consistently low. By fixing recall to be at least 95 %, it produced very low precisions when applied to a number of datasets such as Antihistamines (precision =0 %), SkeletalMuscleRelaxants (precision =0 %), and Triptans (precision =3.65 %).

Adeva et al.'s research (2014) is probably the most comprehensive one so far in this area. They conducted experiments that involved multiple classification algorithms (including naïve Bayes, KNN, Support vector machines, and Rocchio) combined with several feature selection methods (including TF, DF, IDF, etc.) and applied to different parts of the articles (including the titles alone, abstracts alone and both titles and abstracts). SVM has been proved to produce the best performance with respect to the F1 scores. Bekhuis and Demner-Fushman (2012) also compared different algorithms including K-nearest neighbor (KNN), naïve Bayes, complement naïve Bayes (cNB), and evolutionary SVM (EvoSVM) (implemented in the RapidMiner) and used information gain as their feature selection method to select features from article titles and abstract. EvoSVM has been proved to be the most effective among the algorithm. One reason SVM and its variations often outperform other algorithms is that a medical document is normally represented as a feature vector with words or phrases as the features for classification. This feature vector is often high dimensional and sparse; that is, for each document, its feature vector only has a few entries that are non-zero. SVM has the potential to handle large number of features with overfitting protection (Joachims 1998), and it works well with problems with sparse features (Kivinen et al. 1995). Similar to Cohen et al. (2006), Bekhuis and Demner-Fushman's study (2012) also proved the inverse relationship between precision and recall. Precision was maximal when recall was very low, e.g., precision = 100 % and recall = 7.69 %. When maintaining a high recall (100 % for two datasets, ameloblastoma and influenza), evoSVM, though the best among the tested algorithms, produced relatively low precisions (13.11 % for the ameloblastoma dataset and 10.69 % for the influenza dataset).

As mentioned previously, class imbalance remains a critical, yet largely ignored issue in this context. (Shemilt et al. 2013) is perhaps the only research that investigated the use of re-sampling in selecting articles for systematic reviews. They used undersampling by drawing a random sample of excluded records equal in number to the total number records marked as provisionally eligible for inclusion and proved that undersampling helps enhance that the performance of the text-mining based classifiers (Shemilt et al. 2013). In addition to undersampling, oversampling techniques, though never used in the area of systematic reviews, have long been proved to be effective in dealing with class imbalance in data mining literature. For instance, Ling et al. (1998) combined oversampling of the minority class with undersampling of the majority class and concluded that the best results are obtained when both classes are equally represented. A particular type of oversampling, namely the Synthetic Minority Oversampling Technique (SMOTE) (Chawla et al., 2010), creates synthetic examples of the minority class instead of just randomly duplicating minority examples. Chawal et al. (2010) conducted various experiments with different datasets and proved that SMOTE outperforms plain undersampling and oversampling, and furthermore, the combination of SMOTE and undersampling performs even better than SMOTE alone. It is hence intriguing to investigate if re-sampling techniques such as SMOTE can help improve the performance of article classification in the context of systematic reviews.

Overall, the findings of extant research show enough promise to further consider the possibility of using data analytics techniques for automatically screening articles for systematic reviews (Cohen et al. 2006; Frunza et al. 2010; Shemilt et al. 2013; Tsafnat et al. 2014). However, further research is needed to develop appropriate classifiers, resolve the class imbalance problem, and improve the precision of classification techniques while maintaining a high recall.

3 Research gap

Our literature review indicates that 1) for any automated classification technique to be of practical use in supporting article selection for systematic reviews, it is critical for the technique to achieve a high level of recall, and 2) it is necessary to improve precision while sustaining a high recall since a higher precision means that fewer articles would need to be manually reviewed in the downstream full-text triage stage. Improving precision while sustaining a high recall, however, is a difficult task, as shown in existing research. This leads us to the following overarching research question:

1. *How can we develop a classification technique that helps improve precision while sustaining a high recall* (*above 95 %*)?

We plan to address this research question by investigating which combination of textual analytics techniques is most valuable in identifying relevant articles that should be included in a systematic review.

Existing research into automatic article classification for systematic reviews has almost exclusively relied on the bag-of-words approach for feature representation. While this de facto standard has led to promising results, we feel that other feature extraction schemes may provide better predictive ability. Prior research (Liu et al. 2002; Aronson et al. 2007), though not in the area of systematic reviews, has corroborated the observation that biomedical text classification can be improved by enriching raw text with automatically extracted Unified Medical Language System (UMLS) terms. As an example, Kilicoglu et al. (2009) demonstrated the feasibility of automatically identifying "scientifically rigorous" articles using multiple features from publications, including "high-level" features such as Unified Medical Language System (UMLS) terms. This leads us to the following research question:

2. *Can we improve precision while sustaining a high recall by using automatically extracted Unified Medical Language System* (*UMLS*) *terms as features*?

As discussed previously, the issue of class imbalance is critical, yet not sufficiently addressed in this field. To address the issue, Cohen et al. (2006) modified the conventional perceptron algorithm by adjusting the false-negative learning rate (FNLR) to improve the recall to be over 95 %. Another possible approach is using re-sampling methods to re-sample the training data. In the area of data mining, various re-sampling strategies such as undersampling, oversampling and SMOTE oversampling, have been proposed to classify datasets with highly asymmetric positive and negative sample frequency. It is hence meaningful to investigate:

3. *Can we use a re-sampling method to further improve precision while sustaining a high recall*?

4 Methodology

Our analytics approach to identifying relevant articles for systematic reviews includes three major components: 1) feature extraction using the UMLS, 2) soft-margin polynomial SVM, and 3) SMOTE combined with undersampling. We conduct experiments using four systematic review datasets and compared analytics techniques with others that were proposed in existing research. In following sub-sections, we describe the data sources, each component in our analytics approach, and the methods that we compared our techniques with in detail.

4.1 Data sources

We used four systematic reviews on drug topics including ACEInhibitors (ACE), Antihistamines (AN), Skeletal-MusleRelaxants (SKE), and Triptans (TRIP), performed by AHRQ's Evidence-based Practice Center (EPC) at Oregon Health and Science University as our datasets (Cohen 2014). These four systematic review datasets were also used in (Cohen et al. 2006). Cohen et al. (2006) defined a new measure WSS@95 %, i.e., percentage of work saved when recall is fixed to be at least 95 %, to measure the effectiveness of the perceptron-based classifier. The perceptron-based classifier proposed in (Cohen et al. 2006) turned very low WSS@95 % values (0.00 %, 0.00 % and 3.37) and low precisions (3.87 %, 0.00 %, and 3.65 %) on three of the four dataset AN, SKE and TRIP, respectively, when maintaining the recalls to be over 95 %. We hence used these datasets in our experiments since we intended to investigate if our proposed approach can help improve the precision and WSS@95 % values. The perceptron-based classifier achieved relatively high performance (recall =95.61 %; WSS@95 % = 56.61 %) but low precision (3.87 %) for the dataset ACE. We included this dataset in our study to investigate if our approach helps achieve comparable or better WSS@95 % by enhancing precision. The original datasets include the PubMed Unique Identifiers (PMID) of all the

articles and the inclusion and exclusion decisions made by human researchers. Following (Cohen et al. 2006), we focus on classifying the articles based on the title and abstract of the articles. We used Medline's Batch Entrez features to extract the title and abstract of all the articles based their PMIDs. Table 1 shows an overview of the datasets. As discussed above, imbalanced class distributions are the norm for article selection in systematic reviews. Only a small ratio of articles has been included in each of the four systematic reviews. Among the four dataset, SkeletalMuscleRelaxants has the most serious class imbalance problem with only 9 included articles. Consequently, the perceptron-based algorithm proves to be ineffective with precision =0.55 % (classify everything in one class) and WSS@95 % recall (defined later) = 0.00 % for the dataset.

4.2 Feature extraction

We used the MEDLINE records for each article in the four datasets to generate the feature set as input to our classification technique. The feature set includes the features extracted from the title and abstract as well as the article's Medical Subject Headings (MeSH) and MEDLINE publication type. To extract features from the title and abstract of an article, we propose to use the UMLS to automatically extract terms and use them as features. Most of the existing research has relied on the "bag-of-words" approach to extracting features. We conducted experiments to compare the performance between these two methods for feature extraction (i.e., UMLS vs bag-of-words). Below we briefly describe both methods.

The features extracted from the bag-of-words approach used in our comparative experiments included not only unigrams (i.e., individual words) but also 2-term and 3-term n-grams. Each document (i.e., a text file including the article tile and abstract) is represented by a vector of weights m features:

$$d_j = (w_{1j}, w_{2j}, \ldots\ldots, w_{mj})$$

where m is the number of features, and w_i is the weight of the i^{th} features (including unigrams, 2-g and 3-g). The weight value of a feature represents how much that feature contributes to the semantics of the document d_j. If there are n documents in total, the corpus is represented by $n*m$ matrix, which is usually called term-document matrix. In a term-document matrix, if a certain feature (i.e., a word) does not occur in the document, then the weight of that feature becomes 0 for that document. Following (Bekhuis and Demner-Fushman 2012), we used the method TF-IDF(term frequency/inverse document frequency) (Robertson 2004) to create the weights. TF-IDF is a numerical statistic that reveals the importance of a feature in a document in a dataset. The TF-IDF value of a word increases as it appears more often in a document; however, the TF-IDF value is offset by the frequency of the word in the whole dataset. This helps to mitigate for the fact that some words such as "patient" are generally more common than other words in medical documents.

We propose to extract features from the titles and abstracts using the UMLS Metathesaurus. UMLS allows to extract terms from different vocabularies, including CPT, ICD-10-CM, LOINC, MeSH, RxNorm, and SNOMED CT. Moreover, UMLS enables us to extract the Concept Unique Identifier (CUIs), semantic types, and synonymous terms used in medical literature (US National Library of Medicine 2014). We used the MetaMap program that maps words and phrases to different UMLS semantic types. An example of UMLS terms extracted from an abstract is given below.

The free medical text appears as:

> "The objective of this study was to examine the relationships of serum and dietary magnesium (Mg) with prevalent cardiovascular disease (CVD), hypertension, diabetes mellitus, fasting insulin, and average carotid intimal-medial wall thickness measured by B-mode ultrasound."

The UMLS terms and their semantic types appear as:

Study Objective [Idea or Concept]
Relationships [Qualitative Concept]
Serum (Specimen Source Codes - Serum) [Intellectual Product]
Serum (Specimen Type - Serum) [Body Substance]
Dietary Magnesium [Element, Ion, or Isotope]
Cardiovascular (Cardiovascular system) [Body System]
disease prevalence (disorder prevalence) [Quantitative Concept]
Hypertension (Hypertension Adverse Event) [Finding]
Diabetes Mellitus [Disease or Syndrome]
fasting (Act Code - fasting) [Intellectual Product]
Insulin [Amino Acid, Peptide, or Protein, Hormone, Pharmacologic Substance]

Table 1 Overview of Data Corpus

Dataset	Total number of articles	Number of excluded articles	Number of included articles	Ratio—Included vs. Excluded
ACEInhibitors (ACE)	2544	2503	41	1:61
Antihistamines (AN)	310	294	16	1:18
SkeletalMuscleRelaxants (SKE)	1643	1634	9	1:182
Triptans (TRIP)	671	647	24	1:26

Insulin (Recombinant Insulin) [Amino Acid, Peptide, or Protein, Hormone, Pharmacologic Substance]
Average [Quantitative Concept]
Carotid [Body Part, Organ, or Organ Component]
Intima [Tissue]
Medial [Spatial Concept]
Wall (Walls of a building) [Manufactured Object]
Thickness (Thick) [Qualitative Concept]
Measured [Qualitative Concept]
ultrasound b mode (B mode ultrasound) [Diagnostic Procedure]
MEASURED (Measured Tumor Identification) [Diagnostic Procedure]
ultrasound b mode (B mode ultrasound) [Diagnostic Procedure]

We used the UMLS-extracted terms as the features for our classifier. For instance, in the example shown above, the terms such as "Study Objective", "Serum (Specimen Source Codes - Serum)" "Cardiovascular (Cardiovascular system)", "fasting (Act Code - fasting)", etc. have been used as features for classification. In our experiments, we compared the UMLS-based feature extraction method with the conventional bag-of-words approach described above.

4.3 Algorithms

We propose to use soft-margin polynomial SVM to enhance the classification performance and compare it with other algorithms that have proved to be effective in existing research. In order to explain soft-margin polynomial SVM, we describe the regular "hard-margin" SVM algorithm first.

SVM with liner kernel Existing studies such as (Joachims 1998; Liu et al. 2002; Bekhuis and Demner-Fushman 2012) has proved the effectiveness of SVM with a linear kernel in text classification in the process of medical systematic reviews. The optimization problem associated with SVM is shown below.

$$min_{\mathbf{w},b}\frac{\mathrm{w^T w}}{2}$$

$$\text{subject to}: y_i(\mathbf{w}T\mathbf{x_i}+b)\geq 1(\forall \text{ data points } \mathbf{x_i})$$

where for each data point (x_i, y_i), y_i is either 1 or −1, indicating the class to which the point belongs. The two hyperplanes $w \cdot x - b = 1$ and $w \cdot x - b = -1$ are called support vectors that separate the data. SVM maximizes the distance (called "margin") between the support vectors.

Soft-margin polynomial SVM We propose to use the soft-margin Support Vector Machine (SVM) with a polynomial kernel as a classifier. Soft-margin polynomial SVM is an extension of the standard "hard" margin SVM described above.

The "hard-margin" SVM sometimes does not work well since it does not allow data points in the margin. However, data is not often perfectly linearly separable, and it is necessary to allow some data points of one class to appear within the region bounded by the support vectors. Soft-margin polynomial SVM provides the flexibility by introducing a slack variable $\epsilon_i \geq 0$, and the optimization problem of soft-margin polynomial SVM becomes (Stanford 2014):

$$min_{\mathbf{w},b,\epsilon}\frac{\mathbf{w^T w}}{2}+C\sum\nolimits_i \epsilon_i$$

$$\text{subject to}: y_i\left(\mathrm{w}^T\mathrm{x_i}+b\right)\geq 1-\epsilon_i \text{ and } \epsilon_i\geq 0(\forall \text{ data points } \mathbf{x_i}).$$

where ϵ_i, the slack variable, represents the degree of error in classification. The optimization hence becomes a tradeoff between a large margin and a small error penalty (i.e., ϵ_i). When the training set is not linearly separable, and there exists no hyperplane that can perfectly separate positive and negative samples, the optimization results in a "soft" margin that may contain some misclassified data points. The parameter C known as a regularization term can be seen as a method for controlling overfitting - it is tradeoff between the importance of maximizing the margin and fitting the training data. That is, if the C value is large, than model is better fitted to the training data (may cause over-fitting), whereas if the C value is small, SVM fits on the bulk of data (Cortes and Vapnik 1995). In our experiments, when applying soft-margin SVM to each dataset, we selected the best performing C and ϵ value that help maximize precision while sustaining recall to be over 95 %, based on cross-validation.

EvoSVM Bekhuis and Demner-Fushmanb (2012) found that evoSVM achieved best performance, compared with KNN, naïve Bayes, complement naïve Bayes (cNB) (Bekhuis and Demner-Fushman 2012). evoSVM is a SVM implementation using an evolutionary algorithm (ES) to solve the dual optimization problem of a SVM. In our experiments, following Bekhuis and Demner-Fushmanb (2012), we used the RapidMiner's implementation of evoSVM and followed the evoSVM settings recommended by the authors: radial kernel; Gaussian mutation; gamma = 1.0; epsilon =0.1; and C = 1.

Perceptron Cohen et al. (2006) used a perceptron-based classifier to predict when articles should be added to existing drug class systematic reviews. A perceptron is a type of neural network that finds a linear function to discriminate between classes. In essence, a single layer perceptron is simply a linear classifier, which is efficiently trained by a simple rule: It starts with an initial set of guessed weights (i.e. numerical parameters), and then for all wrongly classified data points, the weights either increase or decrease to reduce the prediction errors.

Naïve Bayes Naïve Bayes classifiers are a family of simple probabilistic classifiers based on applying Bayes' theorem with strong (naïve) independence assumptions between the features. According to Adeva et al. (2014), naïve Bayes

seemed to provide the best results in terms of false negatives. We hence also included this algorithm in our comparison.

4.4 Re-sampling methods

We examined four re-sampling techniques for resolving the aforementioned class imbalance issue.

Undersampling reduces the number of samples in the majority class in the training set until the ratio between the minority class and the majority class is at a desired level (Liu et al. 2009). Theoretically, researchers cannot control what information of a majority class is thrown away. Also, undersampling is often problematic since important information about the decision boundary between the majority and minority class may be eliminated (Liu 2004). One of the benefits of undersampling is its very simple implementation. The overall number of samples in a training set is greatly reduced, which means that training time is greatly reduced. In our research, we randomly selected a portion of the majority class, which in our case are the articles excluded from the systems reviews, so that the number of excluded articles in each sampled dataset is equal to that of the included articles. For example, in the ACEInhibitors dataset, there are 1252 excluded articles that were excluded from and 21 included articles. We re-sampled the articles in the training dataset and created a new training set that includes all 21 included articles and 21 randomly selected excluded articles.

Oversampling seeks to increase the number of samples in the minority class by replicating samples from that class (He and Ma 2013). The advantage of this approach is that less information from the majority class is lost, as compared to undersampling. The primary disadvantage of this approach is that it tends to overfit the trained model. In our experiments, we tested different oversampling rates including 100 % (i.e., replicating the minority samples once), 200 % (i.e., replicating the minority samples twice), 300 % (i.e., replicating the minority samples three times), and 400 % (i.e., replicating the minority samples four times). We stopped at 400 % oversampling because our experiments showed that the classifier started to suffer from overfitting on all four datasets. We then select the best performing oversampling rate (among 100 %, 200 %, 300 % and 400 %) based on cross-validation for each dataset.

The Synthetic Minority Oversampling Technique (*SMOTE*) proposed in (Chawla 2010) is different from the conventional oversampling method described above. The conventional oversampling method oversamples the minority class by randomly replicating minority examples. This affects the decision region of the minority class, which results in a similar but more specific region in the feature space (Chawla 2010). In the SMOTE, the minority class is oversampled by creating synthetic examples rather than replicating the minority class examples. In our experiments, we oversampled the minority class by taking each minority class example and developing synthetic examples along the line segments joining any k minority class nearest neighbors (in our case five neighbors). For example, if the rate of oversampling is 200 %, only two neighbors among the five nearest neighbors will be randomly chosen, and a synthetic sample will be generated for each neighbor. If the oversampling rate is 300 %, then for each example in the training dataset, three of its neighbors will be randomly selected, and three synthetic samples will be generated. Synthetic samples are computed according to the following procedure described in (Chawla 2010): 1) compute the difference between the sample under consideration and its nearest neighbor, 2) multiply the difference by a random number between 0 and 1, and 3) add the result from 2) to the feature vector under consideration to create a synthetic sample. We tested SMOTE using different oversampling rates including 100 %, 200 %, 300 % and 400 % to oversample the minority class and selected the best oversampling rate for each dataset based on cross-validation.

A combination of SMOTE and undersampling: We considered combining both SMOTE and undersampling. We investigated combinations of different undersampling rates and SMOTE rates, including 1) 50 % undersampling of the majority class +100 % SMOTE of the minority class, 2) 50 % undersampling of the majority class +200 % SMOTE of the minority class, 3) 75 % undersampling of the majority class +100 % SMOTE of the minority class, 4) 75 % undersampling of the majority class +200 % SMOTE of the minority class, and 5) undersampling of the majority class +200 % SMOTE of the minority class to make the ratio between the majority and minority classes be 1. Again, we selected the best performing combination of sampling rates for each dataset based on cross-validation.

4.5 Evaluation methods

We evaluated the classification performance using four metrics, precision, recall, F1-score and Work Saved over Sampling at 95 % confidence interval or WSS@95 % in short, a metric proposed in (Cohen et al. 2006). These measures are defined based on a confusion matrix as shown in Table 2. In our research, we treated the articles that were included in a review as positive examples and those that were excluded as negative examples. TP represents the number of True Positives, i.e., positive examples that were correctly classified by our SVM classifier. TN is the number of True Negatives,

Table 2 Confusion Matrix

	Predicted Negative	Predicted Positive
Actual Negative	True negative (TN)	False positive (FP)
Actual Positive	False negative (FN)	True positive (TP)

i.e., negative examples that were correctly classified, FP the number of False Positive, i.e., negative examples that were incorrectly classified as positive, and FN the number of False Negatives, i.e., positive examples incorrectly classified as negatives.

The formulas for computing recall, precision, F1 and WSS@95 % are shown in Table 3. Recall refers to the rate of correctly classified positives among all positives and is equal to TP divided by the sum of TP and FN. Precision refers to the rate of correctly classified positives among all examples classified as positive and is equal to the ratio of TP to the sum of TP and FP. F1 means the harmonic mean of recall and precision. WSS@95 % is defined as percentage of examples that meet the initial search criteria and do not need to be manually reviewed because they have been correctly classified. Setting recall above 95 %, WSS can be calculated as the ratio of the sum of TN and FN to the total number of samples minus 0.05.

It is noteworthy that we do not use accuracy or AUC (area under ROC curve) as evaluation metrics for two reasons. First, when the class distribution is imbalanced, the evaluation based on accuracy breaks down. For instance, in the dataset SkeletalMuscleRelaxants, if a classifier classifies all articles (4 positive articles and 817 negative articles) as negative, then the predicted accuracy would be 99.51 %. A very high accuracy rate is achieved without detecting any articles that should be included. Second, classification accuracy assumes equal misclassification costs (for false positive and false negative errors), which is problematic because one type of classification error often can be more expensive than another. In classification for systematic reviews, the cost of false negative is high because we intend to avoid missing any articles that should be included in a systematic review. According to Cohen et al. (2006), any analytics models that achieve a recall less than 95 % is meaningless. Therefore, we preset the recall of a positive class to be greater than 95 %, and we examined approaches to improve the precision of the algorithm. Precision defines the fraction of retrieved documents classified as relevant that are indeed relevant. The higher the precision, the smaller number of articles scientists need to manually review.

To make the most efficient use of the datasets and to get the best estimate of system performance on future data, we chose to follow (Cohen et al. 2006) and used 5 × 2 cross-validation. In 5 × 2 cross-validation, the data set is randomly split in half, and then one half is used to train the classifier, and the classifier is scored using the other half as a test set. Then the roles of the two half data sets are exchanged, with the second half used for training and the first half used for testing, with the results accumulated from both halves of the split (Dietterich 1998). What makes 5 × 2 different from the ten-way cross-validation more commonly used is that the half-and-half split and score process is repeated five times. This approach uses each data sample five times for training and five times for testing among random splits and averages the results together for all runs.

Table 3 Evaluation metrics

Evaluation Metric	Formula
Recall	TP/(TP + FN)
Precision	TP/(TP + FP)
F1	(2*recall*precision)/(recall + precision)
WSS@95 %	(TN + FN)/N – 0.05

N = Total Number of Samples in Positive and Negative Classes.

WSS@95 % = Work Saved over Sampling at 95 % confidence interval.

4.6 Experimental procedures

We conducted two experiments to evaluate the effectiveness of our approach. The datasets we used in the experiments are the four datasets we described in section 4.1 including ACEInhibitors (ACE), Antihistamines (AN), SkeletalMuscleRelaxants (SKE) and Triptans (TRIP). The detail of our experiment design is illustrated in Table 4.

Experiment 1 consists of two steps. First, we used the unigrams, 2-g and 3-g extracted from article titles and abstracts using the bag-of-words approach plus the Medical Subject Headings (MeSH) and MEDLINE publication type as the features and compared soft-margin polynomial SVM with other algorithms including SVM with linear kernel, evoSVM, naïve Bayes, and perceptron. Second, we used the automatically extracted UMLS terms plus the MeSH and MEDLINE publication type as the features. Experiment 1 was designed to compare the performance of soft-margin polynomial SVM against the other algorithms. We also compared the effectiveness of the UMLS-based feature extraction against the bag-of-words method.

After identifying soft-margin polynomial SVM as the most effective algorithm in Experiment 1, we conducted Experiment 2 to investigate if different re-sampling methods including undersampling, oversampling, SMOTE, and SMOTE combined with undersampling can further enhance the classification performance. We also conducted Experiment 2 in two steps. In Step 1, we used features extracted using the bag-of-words approach, and in Step 2, we used the UMLS extracted features. In both steps, we used soft-margin polynomial SVM as the classifier, combined with different re-sampling methods.

5 Experimental results and discussion of findings

5.1 Experiment 1 results

Step 1 In this step we compared multiple algorithms with the features extracted using the bag-of-words approach plus the

Table 4 Overview of experiments

		Features	Algorithms	Sampling method
Exp. 1	Step 1	Bag-of-words (up to 3-g) extracted from titles and abstracts + Medical Subject Headings (MeSH) + MEDLINE publication type	Soft-margin polynomial SVM, SVM, EVO-SVM, Perceptron, Naïve Bayes	N/A
	Step 2	UMLS terms extracted from titles and abstracts + Medical Subject Headings (MeSH) + MEDLINE publication type		
Exp. 2	Step 1	Bag-of-words (up to 3-g) extracted from titles and abstracts + Medical Subject Headings (MeSH) + MEDLINE publication type	Soft-margin polynomial SVM	No sampling Undersampling, Oversampling, SMOTE, SMOTE + Undersampling
	Step 2	UMLS terms extracted from titles and abstracts + Medical Subject Headings (MeSH) + MEDLINE publication type		

MeSH and MEDLINE publication type. The results of this step are shown in Table 5 with the highest performance measures for each dataset being highlighted.

As discussed previously, we intend to improve precision while sustaining a high recall. According to (Cohen et al. 2006), a recall of 0.95 or greater is required for an automated classification system to identify an adequate fraction of the relevant articles. However, among the five algorithms we investigated, two of them, naïve Bayes and evoSVM, do not have sufficient configuration options that allow us to fix recall to be over 95 %. We fixed recall to be at least 95 % for the other three algorithms including soft-margin polynomial SVM, SVM with linear kernel and perceptron. To do so, drawing upon (Cohen et al. 2006), we fixed the false-positive learning rate at 1.0 and adjusted the false-negative learning rate (FNLR) to optimize performance for each dataset. We tested different FNLRs in a consistent manner across for each dataset and applied cross-validation to identify the optimal FNLR that resulted in an as-high-as-possible precision while maintaining over 95 % recall.

Among the three algorithms where we achieved recall of at least 0.95, including soft-margin polynomial SVM, SVM with linear kernel (shown as SVM in Table 5), and perceptron, the soft-margin polynomial SVM was prominent in achieving 100 % recall for three of the four datasets (including AN, SKE and TRIP) and 95.45 % for the dataset ACE. SVM with

Table 5 Experiment 1 step 1 results

Dataset	Algorithm	N	TP	TN	FP	FN	Precision	Recall	F1-sore	WSS@95 %
ACE	Soft-margin SVM	1273	21	809	442	1	4.53	**95.45**	8.65	**58.55**
	SVM	1273	21	81	1171	1	1.76	**95.45**	3.46	1.44
	Perceptron	1273	21	775	476	1	4.23	**95.45**	8.10	55.87
	evoSVM	1273	14	635	617	8	2.22	63.64	4.29	0.00
	Naïve Bayes	1273	7	1245	7	15	**50.00**	31.82	**38.89**	0.00
AN	Soft-margin SVM	156	9	28	119	0	7.03	**100.00**	13.14	**12.95**
	SVM	156	9	16	131	0	6.43	**100.00**	12.08	5.26
	Perceptron	156	0	0	147	9	0.00	0.00	0.00	0.00
	evoSVM	156	4	42	105	5	3.67	44.44	6.78	0.00
	Naïve Bayes	156	2	142	5	7	**28.57**	22.22	**25.00**	0.00
SKE	Soft-margin SVM	809	5	191	613	0	0.81	**100.00**	1.61	**18.61**
	SVM	809	0	804	5	0	0.00	0.00	0.00	0.00
	Perceptron	809	0	0	804	5	0.00	0.00	0.00	0.00
	evoSVM	809	3	318	486	2	0.61	60.00	1.21	0.00
	Naïve Bayes	809	1	803	1	4	**50.00**	20.00	**28.57**	0.00
TRIP	Soft-margin SVM	338	13	107	218	0	5.62	**100.00**	10.64	**26.65**
	SVM	338	13	74	254	0	4.19	**100.00**	8.042	16.89
	Perceptron	338	13	28	297	0	4.19	95.83	8.02	3.28
	evoSVM	338	7	117	208	6	3.26	53.85	6.15	0.00
	Naïve Bayes	338	9	283	42	4	**17.65**	69.23	**28.13**	0.00

linear kernel returned high recalls in three datasets (including ACE, AN, and TRIP), but failed to identify any positive examples, thus resulting in 0 % recall and precision for SKE. The Perceptron algorithm achieved high recalls (95.45.61 % and 95.83 %) for ACE and TRIP, but produced 0 % recall and precision for the other two datasets. Among these three algorithms with fixed recall, soft-margin polynomial SVM achieved the highest precision (4.53 % in ACE, 7.03 % in AN, 0.81 % in SKE, and 5.62 % in TRIP) and the highest F1 scores for all four datasets. Soft-margin polynomial SVM also returned the highest WSS@95 % (56.9 % in ACE, 13 % in AN, 18.6 % in SKE, and 26.65 % in TRIP) for all four datasets. Perceptron returned the second highest WSS@95 % (55.87 %) for the dataset ACE, and SVM with linear kernel returned the second highest WSS@95 % (16.83 %) for TRIP. It is noteworthy that in the case of the dataset SKE, where both SVM with linear kernel and perceptron failed to identify any true positive (TP) examples (see Table 5) and hence returned 0 % precision and 0 % WSS@95 %, soft-margin polynomial SVM was able to produce 18.61 % work reduction. Also, for the dataset AN, soft-margin SVM produced 12.95 % WSS@95 %. It was followed by SVM with linear kernel with a much lower WSS@95 % (5.26 %). Among the four datasets we used, SKE and AN have a smaller number of positive examples (16 and 9 respectively). Our soft-margin SVM appeared to be more effective than the other algorithms in dealing with datasets with a small number of positive articles.

If we consider all of these five algorithms, soft-margin polynomial SVM produced the second highest F1 scores for all four datasets. On the surface, naïve Bayes appeared to have achieved higher precisions and F1 scores. For instance, naïve Bayes returned a high precision (28.57 %) and the highest F1 score (25.00 %) but a low recall (22.22 %) when applied to the dataset AN. However, a close investigation revealed that it returned only two true positive predictions, which means among the nine articles that were included in a systematic review, the naïve Bayes classifier has classified only two of them to be positive. Similarly, for the dataset SKE, naïve Bayes achieved relatively high precision (50 %) and highest F1 score (28.57 %), but made only one true positive prediction. This proves that for asymmetrically distributed datasets, precisions and F-scores are not meaningful when a high recall cannot be obtained. The experimental results in Step 1 clearly showed that among the five algorithms we have compared, soft-margin polynomial SVM achieved the best performance when we used the features extracted using the bag-of-words approach. Moreover, soft-margin polynomial SVM performed significantly better than the other algorithms for the datasets that have a small number of positive examples.

Step 2 In this step, we compared multiple algorithms with features including the automatically extracted UMLS terms plus the Medical Subject Headings (MeSH) and MEDLINE publication type. Table 6 shows the performance of the five algorithms. Again, evoSVM and naïve Bayes returned recall values below the acceptable level (95 %) for all datasets. Among the other three algorithms with over 95 % recall, soft-margin polynomial SVM had the highest precision across all four datasets. It also had 100 % recall for three datasets (AN, SKE and TRIP) and 95.45 % recall for the ACE dataset. SVM with linear kernel produced 95.45 % recall for the ACE dataset, but soft-margin polynomial SVM achieved higher precision (10.14 % vs. 2.72 %) and much higher WSS@95 % (78.74 % vs. 34.36 %). Among the three algorithms with fixed recall, soft-margin polynomial SVM again produced the highest precisions and F1 scores for all of the four datasets. Soft-margin SVM distinguished itself from the other algorithms when applied to the dataset SKE that has only 9 positive examples. While all the other algorithms resulted in 0 % work saved, soft-margin SVM produced 48.89 % WSS. Naïve Bayes had the highest precision and F1 scores; however, the low recalls rendered the precisions and F1 scores hardly meaningful. Our findings in step 2 of experiment 1 are consistent with those obtained in step1. Soft-margin SVM performed better than the other algorithms across all four datasets when we used the automatically extract UMLS terms as the features. It was the optimal algorithm that could provide an improved precision and enhanced percentage of work saved, especially when applied to datasets with few positive examples.

Comparing the results obtained in step 1 vs. step 2, we found that when applied to three datasets including ACE, SKE and TRIP, all three algorithms with recall fixed to be at least 95 % achieved higher precisions, F1 scores and WSS@95 % when UMLS was used to extract features. These three algorithms, however, achieved overall worse results for the dataset AN. Table 7 shows the performance of soft-margin polynomial SVM using the UMLS terms as features vs. using bag-of-words. For the dataset AN, soft-margin SVM successfully identified all included articles in the dataset, but it performed slightly worse with a larger FP value (123 vs. 119), which is not critical given that reviewers just need to manually review 4 additional articles. Using UMLS to extract features significantly enhanced the performance of the soft-margin SVM classifier when applied to the other three datasets. A possible reason behind the UMLS-based feature extraction method outperforming the bag-of-words approach is that the bag-of-words features are created by extracting n-grams from articles without considering the semantics of the words. UMLS (used in conjunction with vocabularies such as CPT, MeSH, SNOMED CT, etc.), on the other hand, identifies the semantic type for each extracted term and provides the synonyms of the term when available. Moreover, using UMLS to extract terms entails an automatic variable selection procedure - it extracts only the terms that are commonly used

Table 6 Experiment 1 step 2 results

Dataset	Algorithm	N	TP	TN	FP	FN	Precision	Recall	F1-sore	WSS@95 %
ACE	Soft-margin SVM	1273	21	1065	186	1	10.41	**95.45**	18.34	**78.74**
	SVM	1273	21	500	751	1	2.72	**95.45**	5.29	34.36
	Perceptron	1273	21	865	386	1	5.16	**95.45**	9.79	63.03
	evoSVM	1273	13	1113	138	9	8.61	59.09	15.03	0.00
	Naïve Bayes	1273	15	1225	26	7	**36.59**	68.18	**47.62**	0.00
AN	Soft-margin SVM	156	9	24	123	0	6.82	**100.00**	12.77	**10.38**
	SVM	156	9	18	129	0	6.52	**100.00**	12.24	0.53
	Perceptron	156	0	147	0	9	0.00	0.00	0.00	0.00
	evoSVM	156	2	137	10	7	16.67	22.22	19.05	0.00
	Naïve Bayes	156	2	138	9	7	**18.18**	22.22	**20.00**	0.00
SKE	Soft-margin SVM	809	5	436	368	0	1.34	**100.00**	2.65	**48.89**
	SVM	809	0	804	0	5	0.00	0.00	0.00	0.00
	Perceptron	809	0	804	0	5	0.00	0.00	0.00	0.00
	evoSVM	809	3	764	40	2	**6.98**	60.00	**12.50**	0.00
	Naïve Bayes	809	2	770	34	3	5.56	40.00	9.76	0.00
TRIP	Soft-margin SVM	329	13	173	152	0	7.88	**100.00**	14.61	**46.18**
	SVM	329	13	122	203	0	6.02	**100.00**	11.35	31.09
	Perceptron	329	13	107	218	0	5.63	**100.00**	10.66	26.66
	evoSVM	329	11	136	189	2	5.50	84.62	10.33	0.00
	Naïve Bayes	329	5	309	16	8	**23.81**	38.46	**29.41**	0.00

in medical literature. This automatic variable selection helps improve classification performance by reducing overfitting.

In summary, the results of experiment 1 demonstrated that 1) soft-margin polynomial SVM consistently performed better than the other algorithms across the four datasets, and 2) overall, using the UMLS terms as features helps enhance the performance of soft-margin polynomial SVM and the other algorithms as well.

5.2 Experiment 2 results

After demonstrating that soft-margin SVM is the better classification algorithm compared with the other algorithms in Experiment 1, we investigated if we can further enhance precision while maintaining a high recall using different re-sampling methods. We tested four re-sampling technique - undersampling, oversampling by replicating minority class examples, SMOTE, and SMOTE combined with undersampling. Again, we conducted the experiment in two steps. In both steps, we used soft-margin SVM as the classifier.

Step 1 In this step, we used the bag-of-words extracted features plus the Mesh and MEDLINE publication type as the features. We compared the four different sampling methods including undersampling, oversampling by replicating minority class examples, SMOTE, and SMOTE combined with undersampling. Table 8 shows the results obtained in this step. It also includes the performance measures of soft-margin

Table 7 Comparing soft-margin SVM results obtained in step 1 vs those obtained in step 2

Dataset	Feature extraction method	N	TP	TN	FP	FN	Precision	Recall	F1-sore	WSS@95 %
ACE	Bag-of-words	1273	21	789	463	1	4.34	95.45	8.30	56.90
	UMLS	1273	21	1065	186	1	**10.41**	**95.45**	**18.34**	**78.74**
AN	Bag-of-words	156	9	28	119	0	**7.03**	**100.00**	**13.14**	**13.06**
	UMLS	156	9	24	123	0	6.82	100.00	12.77	10.38
SKE	Bag-of-words	809	5	191	613	0	0.81	100.00	1.61	18.6
	UMLS	809	5	436	368	0	**1.34**	**100.00**	**2.65**	**48.89**
TRIP	Bag-of-words	338	13	107	218	0	3.63	100.00	10.66	26.65
	UMLS	338	13	173	152	0	**7.88**	**100.00**	**14.61**	**46.18**

Table 8 Experiment 2 step 1 results with features extracted based on bag-of-words

Dataset	Sampling method	N	TP	TN	FP	FN	Precision	Recall	F1-sore	WSS@95 %
ACE	Undersampling	1273	21	859	392	1	5.08	**95.45**	9.66	62.56
	Oversampling	1273	21	853	398	1	5.01	**95.45**	9.52	62.09
	SMOTE	1273	21	952	299	1	6.56	**95.45**	12.28	69.86
	SMOTE + Undersampling	1273	21	981	270	1	**7.22**	**95.45**	**13.42**	**72.14**
	Non-sampling	1273	21	809	442	1	4.53	**95.45**	8.65	58.55
AN	Undersampling	156	9	4	143	0	5.92	**100.00**	11.18	0.00
	Oversampling	156	9	21	126	0	6.67	**100.00**	12.50	8.46
	SMOTE	156	9	22	125	0	6.72	**100.00**	12.59	9.10
	SMOTE + Undersampling	156	9	21	126	0	6.67	**100.00**	12.50	8.46
	Non-sampling	156	9	28	119	0	**7.03**	**100.00**	**13.14**	**12.95**
SKE	Undersampling	809	5	107	697	0	0.71	**100.00**	1.41	8.23
	Oversampling	809	5	317	487	0	1.02	**100.00**	2.01	34.18
	SMOTE	809	5	434	370	0	**1.33**	**100.00**	**2.63**	**48.65**
	SMOTE + Undersampling	809	5	400	404	0	1.22	**100.00**	2.42	44.44
	Non-sampling	809	5	191	613	0	0.81	**100.00**	1.61	18.61
TRIP	Undersampling	338	13	43	282	0	4.41	**100.00**	8.44	7.72
	Oversampling	338	13	134	191	0	6.37	**100.00**	11.98	34.64
	SMOTE	338	13	164	161	0	7.47	**100.00**	13.90	43.52
	SMOTE + Undersampling	338	13	201	124	0	**9.49**	**100.00**	**17.33**	**54.47**
	Non-sampling	338	13	107	218	0	5.62	**100.00**	10.64	26.65

SVM when no re-sampling has been conducted (shown as "non-sampling" in Table 8).

Undersampling means that we randomly select a subset of the negative examples (articled excluded from the systematic reviews in this case), so that the number of positive examples is equal to that of the positive examples. When compared with non-sampling, undersampling was only able to produce the improved performance for the dataset ACE (62.5 % WSS@95). It failed to achieve improved performance for both SKE and TRIP. Undersampling did not work at all for the dataset AN. It helped to improve the classification performance for the dataset ACE due to the fact that there are relatively a large number of positive examples, which might be sufficient to train the classifier. We then oversampled the minority class examples (i.e., the included articles). For each dataset, we selected the optimal sampling rate based on the method described in section 4.4. Oversampling by replicating the minority class examples (shown as "oversampling" in Table 8) enhanced classification performance with respect to the F1 score and WSS@95 % for three datasets including ACE, SKE and TRIP. It worked especially well for the dataset SKE with only 9 positive examples. SMOTE is another oversampling technique for increasing the number of minority class examples. Compared with non-sampling, SMOTE showed significantly improved performance for two datasets SKE and TRIP. It boosted WSS@95 % from 18.61 % to 48.65 % for SKE and from 26.65 % to 43.52 % for TRIP. As shown in Table 9, SMOTE also outperformed plain oversampling across all four datasets. Combining SMOTE and under-sampling enabled our classifier to achieve higher precisions, F1 scores and WSS@95 % than SMOTE alone for two datasets including ACE and TRIP. It produced slightly worse performance for the other two datasets. The datasets ACE and TRIP have larger numbers of included articles than the other two datasets, which indicates that with the bag-of-words features, SMOTE combined with undersampling may be the optimal re-sampling method when applied to datasets with relatively a large number of positive examples, while we may need to use SMOTE alone when dealing with datasets with a small number of positive examples. It is also noteworthy that for the dataset AN, the classifier without any re-sampling achieved the best performance.

Step 2 In this step, we used the UMLS terms as the features. Again, we compared the four different sampling methods including undersampling, plain oversampling, SMOTE oversampling, and SMOTE combined with undersampling. Table 9 shows the results we obtained in this step.

With the UMLS terms as the features, the classifier with undersampling showed performance that is consistent with what we obtained in Step 1. It did not work at all for the dataset AN. Compared with non-sampling, undersampling failed to improve performance for three datasets except ACE. Different form the results we obtained from Step 1,

Table 9 Experiment 2 step 2 results

Dataset	Sampling method	N	TP	TN	FP	FN	Precision	Recall	F1-sore	WSS@95 %
ACE	Undersampling	1273	21	1065	186	1	10.82	**95.45**	19.44	78.74
	Oversampling	1273	21	936	315	1	6.50	**95.45**	12.17	68.61
	SMOTE	1273	21	1096	155	1	12.88	**95.45**	22.70	81.17
	SMOTE + undersampling	1273	21	1104	147	1	**13.55**	**95.45**	**23.73**	**81.80**
	Non-sampling	1273	21	960	264	1	7.72	**95.45**	14.29	70.49
AN	Undersampling	156	9	6	141	0	6.00	**100.00**	11.32	0.00
	Oversampling	156	9	22	125	0	6.72	**100.00**	12.59	9.10
	SMOTE	156	9	38	109	0	7.63	**100.00**	14.17	19.36
	SMOTE + undersampling	156	9	43	104	0	**7.96**	**100.00**	**14.75**	**22.56**
	Non-sampling	156	9	24	123	0	6.82	**100.00**	12.77	10.38
SKE	Undersampling	809	5	349	455	0	1.15	**100.00**	2.28	38.14
	Oversampling	809	5	516	342	0	1.56	**100.00**	3.08	58.78
	SMOTE	809	5	478	326	0	1.64	**100.00**	3.24	54.09
	SMOTE + undersampling	809	5	630	174	0	**3.29**	**100.00**	**6.37**	**72.87**
	Non-sampling	809	5	436	368	0	1.45	**100.00**	2.85	48.89
TRIP	Under-sampling	338	13	62	263	0	4.78	**100.00**	9.12	13.34
	Oversampling	338	13	204	121	0	10.00	**100.00**	18.18	55.36
	SMOTE	338	13	215	110	0	10.92	**100.00**	19.70	58.61
	SMOTE + under-sampling	338	13	220	105	0	**11.40**	**100.00**	**20.47**	**60.09**
	Non-sampling	338	13	173	152	0	8.07	**100.00**	14.94	46.18

for AN, both SMOTE alone and SMOTE combined with undersampling produced better precision and WSS@95 % values than non-sampling. It is noteworthy that SMOTE combined with undersampling appeared to be the best re-sampling method for all four datasets. It worked particularly well for the dataset SKE with only 9 positive examples. It doubled the precision produced by SMOTE alone and raised the WSS@95 % value from 54.09 % to 72.87 %.

In Table 10, we compared the best performing re-sampling methods obtained in Step1 and in Step 2. With the automatically extracted UMLS terms as the features in Step 2, SMOTE combined with under-sampling achieved better performance for all four datasets, and it worked particularly well for AN and SKE.

In summary, the results of experiment 2 demonstrated that 1) overall, SMOTE-based re-sampling methods including both SMOTE alone and SMOTE combined with undersampling helped improve classification performance of the soft-margin SVM classifier, whether we used the UMLS extracted features or bag-of-words; 2) the combination of SMOTE and undersampling in general performed better than SMOTE alone when the UMLS terms were used as the features. It is understandable that undersampling failed to achieve high performance since in undersampling, we make the ratio between the positive class and the negative class equal to 1 by reducing the number of negative examples, thus losing considerable amounts of information from the negative examples. SMOTE in general outperformed plain oversampling because in plain oversampling, the decision region that results from classification of the minority class actually becomes smaller as we replicate the minority class examples. SMOTE offers more related minority class examples to learn from, which leads to more coverage of the minority class, thus allowing a learner to create broader decision regions (Chawla 2010). Moreover, oversampling tends to cause overfitting because of repetitive instances that tightens the decision boundary. In contrast, with artificially created examples, SMOTE softens the boundary region and is hence less susceptible to overfitting (Longadge et al. 2013).

Finally, following suggested data mining practice (Liu et al. 2007), we compared our analytics techniques with an existing benchmark model. The benchmark we used is the perceptron model developed in Cohen et al.'s study (2006), a NIH-funded project that represents one of the most significant research in this field. Although Cohen et al. used the bag-of-words method to extract the features and did not employ any re-sampling methods, these two studies are comparable since we used the same datasets, the same data sources (including titles, abstracts, MeSH, and MEDLINE publication type) in each dataset to extract features, and the same evaluation metrics (including precision, recall, F1 score and WSS@95 %). Figure 1 shows the comparison of our proposed method with the benchmark model.

Table 10 Comparing soft-margin SVM results obtained in step 1 vs those obtained in step 2

Dataset	Step	Best sampling method	N	TP	TN	FP	FN	Precision	Recall	F1-score	WSS@95 %
ACE	1	SMOTE + Undersampling	1273	21	981	270	1	7.22	95.45	13.42	72.14
	2	SMOTE + undersampling	1273	21	1104	147	1	**13.55**	**95.45**	**23.73**	**81.80**
AN	1	Non-sampling	156	9	28	119	0	7.03	100.00	13.14	12.95
	2	SMOTE + undersampling	156	9	43	104	0	**7.96**	**100.00**	**14.75**	**22.56**
SKE	1	SMOTE	809	5	434	370	0	1.33	100.00	2.63	48.65
	2	SMOTE + undersampling	809	5	630	174	0	**3.29**	**100.00**	**6.37**	**72.87**
TRIP	1	SMOTE + Undersampling	338	13	201	124	0	9.49	100.00	17.33	54.47
	2	SMOTE + under-sampling	338	13	220	105	0	**11.40**	**100.00**	**20.47**	**60.09**

As shown in Fig. 1, our approach that includes a combination of different text analytics techniques produced higher recalls, precisions, and F1-scores over all four datasets, compared with the benchmark model. The significantly improved WSS values indicate that our approach significantly reduced the number of articles that scientists need to manually review to develop systematic reviews, thereby having the potential to reduce labor and other costs associated with systematic reviews. Our proposed approach worked especially well for the datasets AN and SKE, each of which has only a few included article. For example, our approach produced 72.87 % WSS@95 % for the dataset SKE. Reviewers initially queried and included 809 documents in the dataset SKE. A manual process will entail reviewing all 630 documents to end up with five relevant documents. In contrast, our proposed approach would have accurately removed 174 documents. This leaves only 152 articles for the reviewers to manually review (resulting in the five relevant articles).

6 Conclusion

Evidence-based medicine has been widely promoted as a means of improving clinical outcomes, where evidence-based medicine refers to the practice of medicine based on the best available scientific evidence. Information overload, however, makes it difficult for healthcare providers to easily integrate evidence into practice. The challenge not only lie in recognizing the potential for breakthroughs in health but in *realizing* this potential by providing the right tools to find the data that are relevant to you, extract information from the data, and convert that information to actionable

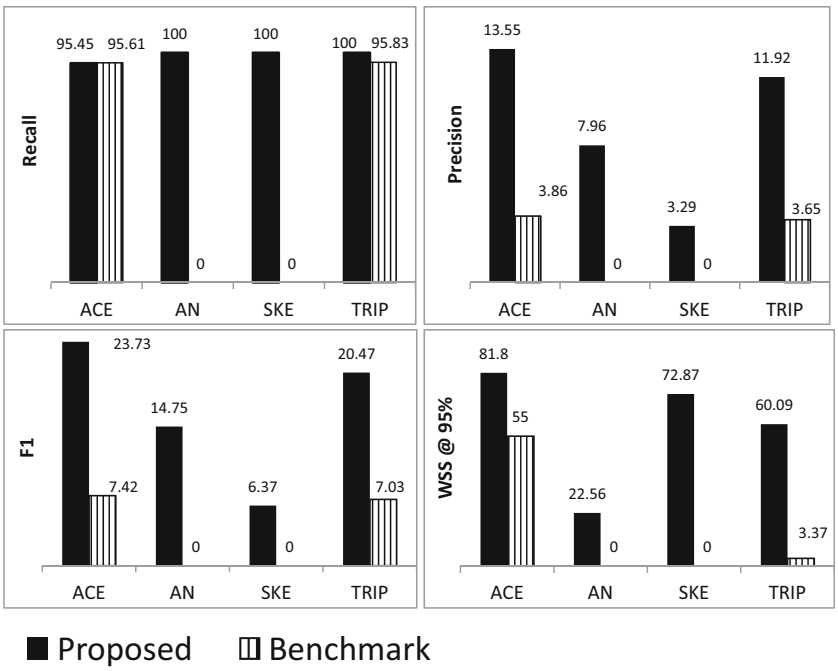

Fig. 1 Comparison of proposed model with the benchmark model

knowledge. Information technology (IT) plays a crucial role in the practice of evidence-based medicine (EBM) by allowing health care practitioners to access and evaluate clinical evidence as they formulate their patient care strategies (Wells 2006). This oftentimes involves an analysis of a large amount of complex information.

This research focuses on systematic reviews, the heart of evidence-based medical practice (Stevens 2001). The creation and update of these reviews is resource intensive. A major bottleneck occurs when scientists screen medical studies. Scientists need to identify provisionally eligible studies by reading the title and abstract of thousands of articles. This challenge calls for the use of text analytics to automate the article selection process. In this research, we examined an automated method to classify relevant articles for inclusion or exclusion during the abstract triage stage for creating and updating systematic reviews of medical research. We demonstrated that a novel combination of text analytics techniques, including using the automatically extracted UMLS terms as the features, soft-margin polynomial SVM as the classification algorithm and SMOTE combined with undersampling to deal with the class balance issue, help improve precision while sustaining a high recall (95 % or higher) in article classification for SRs.

Our research is intended to make the following contributions. From a theoretical perspective, this research explores the possibility of combining different text analytics techniques in the area of systematic review development. In prior research, the bag-of-words method has been used as the de facto standard methods for extracting features from article titles and abstract. We used the automatically extracted UMLS terms as feature by leveraging the latest version of the MetaMap software and demonstrated that this feature extraction method helps enhance classification performance, as compared with the bag-of-words approach. The class imbalance issue has been insufficiently addressed in extant literature. We explored the use of various re-sampling methods, which have been hardly used in this field, to alleviate the class imbalance problem. We modified SMOTE by combining it with undersampling and used it to enhance article classification performance. The experiences and lessons learned from this research are expected to inform the literature regarding the efficacy of the proposed techniques and the further development and refinement of these techniques.

From a practical and applied research perspective, this research is expected to result in a significant reduction in the cost of creating and updating systematic reviews. Currently, in the context of medical knowledge generation, the substantial cost of selecting articles for systematic reviews precludes us from creating and updating systematically reviews to keep pace with medical research advances, which subsequently impedes the translation of the latest medical evidence into healthcare practice. This research can help automate the systematic review development process by significantly reducing the number of articles scientists need to manually review and has the potential to contribute to the adoption of evidence-based medicine. In summary, this research provides direct impact in the availability of best medical evidence, and consequently, may contribute to improving the health and wellbeing of society.

The research can be further extended along a number of dimensions. First, the proposed approach can be further evaluated using additional data sets (beyond the four sets included in this research). Second, this approach can be extended to support the creation of systematic reviews. The current approach is more suited to updating existing systematic reviews where there is already a pre-classified dataset that can be used for learning purposes. Last but not least, future research may investigate means for deploying the proposed approach in a manner that simplifies and automates (or semi-automate) the update of systematic reviews on a frequent basis as new literature is added to the existing knowledge repository. Other integration and deployment possibilities include the leverage of clinical trials documentation, e.g., from clinicaltrials.gov to further expedite the translation of medical research into practice.

In conclusion, this research further attests to the potential of machine learning, text mining and big data analytics in supporting evidence-based medicine. It is a step towards closing the gap between research and practice in the quest towards providing higher quality healthcare outcomes at a reduced cost.

References

Adeva, G., Atxa, P., Carrillo, U., & Zengotitabengoa, A. (2014). Automatic text classification to support systematic reviews in medicine. *Expert Systems with Applications*, *41*(4), 1498–1508.

Allen, I., & Olkin, I. (1999). Estimating time to conduct a meta-analysis from number of citations retrieved. *JAMA*, *282*(7), 634–635.

Ananiadou, S., Procter, R., Rea, B., & Sasaki, Y. (2009). *Supporting Systematic Reviews Using Text Mining.*, 3.

Aronson, A. R., Bodenreider, O., Demner-Fushman, D., Fung, K. W., Lee, V. K., Mork, J. G., et al. (2007) From indexing the biomedical literature to coding clinical text: experience with MTI and machine learning approaches. In *Proceedings of the Workshop on BioNLP 2007*: *Biological*, *Translational*, *and Clinical Language Processing*, *2007* (pp. 105–112): Association for Computational Linguistics

Bekhuis, T., & Demner-Fushman, D. (2012). Screening nonrandomized studies for medical systematic reviews: a comparative study of classifiers. *Artificial Intelligence in Medicine*, *55*, 197–207. doi:10.1016/j.artmed.2012.05.002.

Chawla, N. V. (2010). Data mining for imbalanced datasets: an overview. *Data mining and knowledge discovery handbook, Springer.*

Cochrane (2013). Cochrane handbook for systematic reviews of interventions. http://handbook.cochrane.org. Accessed Nov 20, 2013.

Cohen, A. M. C. (2014). Systematic drug class review gold standard data. http://skynet.ohsu.edu/~cohenaa/systematic-drug-class-review-data.html. Accessed April 2, 2014.

Cohen, A., Ersh, W., & Eterson, K. (2006). Reducing workload in systematic review preparation using automated citation classification. 206–219, doi:10.1197/jamia.M1929.The.

Cohen, A., Adams, C., Davis, J., Yu, C., Yu, P., Meng, W., et al. (2010). The Essential role of systematic reviews, and the need for automated text mining tools. 376–380.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297.

Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, *10*(7), 1895–1923.

Frunza, O., Inkpen, D., & Matwin, S. (2010). Building systematic reviews using automatic text classification techniques. *Proceedings of the 23rd International Conference on Computational Linguistics*: *Posters. Association for Computational Linguistics*, 303–311.

He, H., & Ma, Y. (2013). *Imbalanced Learning*: *Foundations*, *Algorithms*, *and Applications*: Technology & engineering.

Higgins, J., & Green, S. (2011). Cochrane Handbook for Systematic Reviews of Interventions Version 5.1.0 [updated March 2011]. *The Cochrane Collaboration*.

Joachims, T. (1998). Text categorization with support vector machines : learning with many relevant features. *Universtat Dortmund*, *1-19*.

Kilicoglu, H., Demner-Fushman, D., Rindflesch, T. C., Wilczynski, N. L., & Haynes, R. B. (2009). Towards automatic recognition of scientifically rigorous clinical research evidence. *Am Med Inform Assoc*, *16*(1), 25–31. doi:10.1197/jamia.M2996.

Kivinen, J., Warmuth, M., & Auer, P. (1995). The perceptron algorithm vs. winnow: Linear vs. logarithmic mistakes bounds when few input variables are relavant. *Conference on Computational Learning Theory*.

Liu, A. Y. (2004). *The effect of oversampling and undersampling on classifying imbalanced text datasets*. The University of Texas at Austin.

Liu, H., Johnson, S. B., & Friedman, C. (2002). Automatic resolution of ambiguous terms based on machine learning and conceptual relations in the UMLS. [Evaluation Studies

Liu, T. Y., Xu, J., Qin, T., Xiong, W., & Li, H. (2007). Letor: Benchmark dataset for research on learning to rank for information retrieval. *In Proceedings of SIGIR 2007 workshop on learning to rank for information retrieval*, 3–10.

Liu, X. Y., Wu, J., & Zhou, Z.-H. (2009). Exploratory undersampling for class-imbalance learning. *IEEE Transactions On SYSTEMS, Man, And Cybernetics—Part B: Cybernetics*, *39*(2), 539–550.

Matwin, S., Kouznetsov, A., Inkpen, D., Frunza, O., & O'Blenis, P. (2010). A new algorithm for reducing the workload of experts in performing systematic reviews. [research support, Non-U.S. Gov't]. *Journal of the American Medical Informatics Association*, *17*(4), 446–453. doi:10.1136/jamia.2010.004325.

McGowan, J., & Sampson, M. (2005). Systematic reviews need systematic searchers. *Journal of the Medical Library Association*, *93*(1), 74–80.

Mulrow, C. (1994). Rationale for systematic reviews. *BMJ*, *309*, 597–599.

Research Support, U.S. Gov't, P.H.S.]. *J Am Med Inform Assoc*, *9*(6), 621–636.

Robertson, S. (2004). Understanding inverse document frequency: on theoretical arguments for IDF. *Journal of Documentation*, *60*(5), 503–520.

Shemilt, I., Simon, A., Hollands, G. J., Marteau, T. M., Ogilvie, D., O'Mara-Eves, A., et al. (2013). Pinpointing needles in giant haystacks: use of text mining to reduce impractical screening workload in extremely large scoping reviews. *Research Synthesis Methods, n/a-n/a*. doi:10.1002/jrsm.1093.

Shojania, K. G., Sampson, M., Ansari, M. T., & Garritty, C. (2007a). Updating systematic reviews. *AHRQ*, *16*.

Shojania, K. G., Sampson, M., Ansari, M. T., Garritty, C., Doucette, S., Rader, T., et al. (2007b). Updating Systematic Reviews. *Agency for Healthcare Research and Quality*, *Contract No. 290–02–0021*.

Stanford (2014). Soft margin classification. http://nlp.stanford.edu/IR-book/html/htmledition/soft-margin-classification-1.html. Accessed June 11, 2014.

Stevens, S. (2001). Systematic reviews: the heart of evidence-based practice. *AACN Clinical Issues: Advanced Practice in Acute & Critical Care*, *12*(4), 529–538.

Tsafnat, G., Glasziou, P., Choong, M. K., Dunn, A., Galgani, F., & Coiera, E. (2014). Systematic review automation technologies. *Syst Rev*, *3*, 74. doi:10.1186/2046-4053-3-74.

US National Library of Medicine (2014). Unified Medical Language System® (UMLS®). http://www.nlm.nih.gov/research/umls/knowledge_sources/metathesaurus/index.html2014.

Wallace, B. C., Trikalinos, T. a., Lau, J., Brodley, C., & Schmid, C. H. (2010). Semi-automated screening of biomedical citations for systematic reviews. *BMC Bioinformatics*, *11*, 55. doi:10.1186/1471-2105-11-55.

Wells, S. Role of information technology in evidence based medicine: advantages and limitations (2006). *The Internet Journal of Healthcare Administration,* 4, 2.

Prem Timsina is a doctoral student at Dakota State University. He obtained his undergraduate in computer engineering from Tribhuvan University, Nepal and Lumiere University, France. His research interests include machine learning, big data, data and text mining, health analytics, and leveraging data analytics to support business intelligence and decision-making. He have published several articles in international journals like International Journal of Medical Informatics, and given various talks in conferences like Americas Conference on Information Systems, and Hawaii International Conference on System Sciences.

Jun Liu is an assistant professor in information systems in the College of Business & Information System, Dakota State University. He obtained his Ph.D. and M.Sc. in Management Information Systems from the Eller College of Management, University of Arizona. His research interests include data and text mining, social network analysis, data provenance, examining user collaboration in open source environments such as Wikipedia, and using technology to support business intelligence and decision-making. He has published several research papers in internationally refereed journals such as ACM Transactions on Management Information Systems, Journal of Data Semantics, Journal of Computing Science and Engineering, International Journal of Intelligent Information Technologies, Lecture Notes in Computer Science, etc. and has presented several papers at several international conferences.

Omar El-Gayar is a Professor of Information Systems and Dean of Graduate Studies and Research, Dakota State University. His research interests include: analytics, business intelligence, and decision support with applications in problem domain areas such as healthcare, environmental management, and security planning and management. His interdisciplinary educational background and training is in information technology, computer science, economics, and operations research. Dr. El-Gayar's industry experience includes working as an analyst, modeler, and programmer. His numerous publications appear in various information technology related fields. He is a member of AIS, ACM, INFORMS, and DSI.