Journal of Biomedical Informatics 43 (2010) 6-December, pp. 902-913

DOI: 10.1016/j.jbi.2010.07.010

A comparison of machine learning techniques for detection of drug target articles

Roxana Danger [a], Isabel Segura-Bedmar [b,*], Paloma Martínez [b], Paolo Rosso [a]

[a] *Natural Language Engineering Lab. – ELiRF. Dpto. de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia, Valencia, Spain*
[b] *Dpto. de Informática, Universidad Carlos III de Madrid, Leganés, Madrid, Spain*

Abstract: Important progress in treating diseases has been possible thanks to the identification of drug targets. Drug targets are the molecular structures whose abnormal activity, associated to a disease, can be mod-ified by drugs, improving the health of patients. Pharmaceutical industry needs to give priority to their identification and validation in order to reduce the long and costly drug development times. In the last two decades, our knowledge about drugs, their mechanisms of action and drug targets has rapidly increased. Nevertheless, most of this knowledge is hidden in millions of medical articles and textbooks. Extracting knowledge from this large amount of unstructured information is a laborious job, even for human experts. Drug target articles identification, a crucial first step toward the automatic extraction of information from texts, constitutes the aim of this paper. A comparison of several machine learning techniques has been performed in order to obtain a satisfactory classifier for detecting drug target articles using semantic information from biomedical resources such as the Unified Medical Language System. The best result has been achieved by a Fuzzy Lattice Reasoning classifier, which reaches 98% of ROC area measure.

Keywords: Biomedical text classification, Biomedical, information retrieval, Drug discovery, Drug target, Machine learning, Support Vector Machines, Naïve Bayes, Unified Medical Language System, MetaMap.

1. Introduction

A drug target is defined as a molecular structure within the organism, that is linked to a disease, and whose activity is either stimulated or inhibited by drugs that are administered to fight or diagnose the said disease [1]. Several studies have tried to estimate the total number of drug targets [1,2], however, no consensus has been reached yet. While some studies [1] estimate that current target counts are of the order of 100, other suggest a higher order of magnitude ([3] reported 14,000 targets).

In recent years, important progress in treating diseases such as cancer, AIDS, or Parkinson's disease, among many others, has been possible thanks to the identification of drug targets linked to these diseases [4–6]. The current drug discovery process is mainly focused on the search and validation of drug candidates that act on a particular therapeutic target [7]. Firstly, the process of a particu-lar disease is studied and its physiologic mechanisms are deter-mined to detect the drug targets related to this disease. Then, new drugs are designed to act on these targets. Due to the high cost and the long time required by the drug development process, pharmaceutical industry needs to improve the strategies for prioritizing targets and drug candidates in the drug discovery process. A broader knowledge of these targets can help to understand the mechanisms of action of drugs at molecular level and provide insights that guide drug design and the search for new targets.

As a consequence of the above, new research studies on drug targets are continually published [8–10]. In addition, during the last years there has been a growing interest in the development of useful knowledge resources about drug targets. The Therapeutic Target Database (TTD) [11] was developed to provide public and accessible information about 1535 protein and nucleic acid targets reported in the literature, their targeted disease conditions, and the drugs that act on each of these targets. Recently, the Drug Target Prioritization Network, established by the World Health Organization (WHO), has developed the Drug Target Tropical Disease Research (TDR[1]) Prioritization Database [12], a new online resource to integrate genomic information relevant for drug discovery on pathogens that cause human infectious diseases. The aforementioned resources can facilitate researchers in looking for information on possible targets, and consequently, they can have an important impact on the opening of new ways for drug discovery. However, the main problem of these resources is that their manual construction is a time-consuming, labor-intensive and expensive task.

Despite the availability of a growing amount of structured pharmacological, biological, genetic and medical information, most of this information is unstructured, hidden in millions of medical articles and textbooks, and accessible only to human specialists. Furthermore, knowledge on drug targets is far from being com-

* Corresponding author.
E-mail address: isegura@inf.uc3m.es (I. Segura-Bedmar).

[1] http://TDRtargets.org.

plete since there are efficient drugs whose molecular effects are still unknown on the human metabolism [1].

Manual management and analysis of the large amount of textual information in this field is an infeasible task. The overwhelming number of publications makes it impossible to keep up-to-date with the recent and relevant developments in the biomedical domains. Extracting knowledge from this large amount of unstructured information is a laborious job, even for human experts. Therefore, a challenging goal for improving the efficiency of the drug discovery process is to develop automated systems that aid researchers managing this large amount of publications.

To our knowledge, only one approach has addressed the semiautomatic data-base curation of drug–target interactions. In the SuperTarget[2] [13] database, the efforts for drug target annotation were reduced by the use of the text mining tool EbiMed [14]. This tool retrieves abstracts by querying keywords from MedLine and filters sentences that contain at least two biomedical entities. EBIMed labels a protein name if it co-occurs with another protein, gene, drug or species name. In order to recognize these biomedical terms, EbiMed uses a set of bioinformatics resources: UniProtKB/Swiss-Prot [15], MedLinePlus web site[3], Gene Ontology [16] and the NCBI taxonomy[4]. Subsequently, the list of real relations was assembled by manual curation.

Machine learning techniques are currently used for classification tasks, and in this work we apply them for detecting articles that contain drug–target interactions, in order to reduce the time and effort needed to manually curate a drug–target database. In this paper, a variety of machine learning techniques have been applied to the classification of drug target relevant articles in order to obtain a satisfactory classifier. The approach is evaluated in the context of a binary classification of documents. This binary classification can correspond to a stage in the information retrieval process where the possible relevant documents are selected from the mass of non-relevant ones before being more thoroughly examined later on.

In addition, we believe that UMLS Metathesaurus [17], a comprehensive ontology that integrates a wealth of biomedical terminological resources, may be more comprehensive and robust than the resources used by EbiMed. We hypothesize that the semantic information obtained from biomedical resources such as UMLS or MeSH (Medical Subject Headings) [18] index can benefit the classification of documents because of the possibility of reducing the sparseness of data.

The paper is organized as follows: Section 2 reviews the related works. Section 3 describes our proposal. Section 4 presents the evaluation framework of our approach and the results we have obtained. Section 5 presents conclusions and future works.

2. Related work

The task we are facing requires knowledge about available biomedical information resources, suitable solutions for biomedical text mining problems, and biomedical text classification tools. These three themes are the subjects of the following subsections.

2.1. Biomedical information resources

Life science disciplines are prolific producers of massive amounts of information distributed in a huge number of bibliographical and terminological knowledge resources. Although a comprehensive review of these resources is out of the scope of this paper, this section provides an outline of the main resources used by our proposal.

MedLine is a bibliographic database covering several biological and bio-medical fields with about 18 million references of journal articles. PubMed[5] is an online service that provides public access to Medline. MeSH is a hierarchy of medical terms that is used to index articles included in MedLine. Each Medline article is manually associated to a set of MeSH concepts which characterizes it. Thus, MeSH provides a consistent way to deal with the terminological variability problem which may adversely affect the retrieval information process. MeSH is part of the Unified Medical Language Systems (UMLS) whose main objective is to assist in the developing of natural language technology for biomedical texts. UMLS has three major knowledge sources: the Metathesaurus, the Semantic Network and the Specialist Lexicon. The MetaMap Transfer (MMTx) program [19] analyzes the texts syntactically and selects the concepts of the UMLS Metathesaurus that best fit a certain phrase.

DrugBank [3,20] is an annotated database with about 4900 drug entries. Each entry contains more than 100 data fields that gather detailed chemical and pharmacological information (type, category, brand name, chemical formula, drug interactions, etc.). Regarding the drug target information contained in DrugBank, each drug is related to one or more drug targets. DrugBank's list of drug targets has been manually compiled from several drug targets sources such as TTD or the list provided by [1]. DrugBank also contains a set of MedLine article references for each drug target.

2.2. Text mining tools for biomedical information retrieval

Recently, Bioalma, a Spanish IT company specialized in the research and development of biomedical software, has launched NovoSeek[6], a tool that may be serve as a search engine alternative to PubMed. NovoSeek ranks the retrieved documents according to biomedical concepts such as diseases, drugs, genes, among others. In addition, this tool helps users to improve their queries by the use of synonyms.

EBIMed [14] is a service developed by the European Bioinformatics Institute (EBI) to retrieve information from MedLine. As it was mentioned in the Introduction, this tool combines document retrieval with co-occurrence-based analysis of MedLine abstracts. EBIMed has been mainly focused on improving the access to information about protein–protein interactions and effects of drugs on proteins (drug targets).

iHOP (information Hyperlinked Over Proteins) [21] is a web service that automatically extracts key sentences from MedLine documents. Genes, proteins and chemical compounds terms are annotated and linked to MeSH terms by machine learning methods.

2.3. Biomedical text classification

In recent years, several competitions such as KDD 2002 Challenge Cup [22], TREC Genomics Track or BioCreAtIvE (Critical Assessment for Information Extraction in Biology) Challenges have promoted research on text classification methods in the biomedical domain, since they provide a suitable framework and datasets for evaluating and comparing different approaches.

KDD 2002 Cup focused on identifying what papers contain experimental evidence for Drosophila gene expression. TREC 2004 and 2005 Genomics Tracks, [24,25], pursued the classification of full-text documents simulating the task of curators for the Mouse Genome Informatics (MGI)[7] database [23]. In both tracks,

[2] http://insilico.charite.de/supertarget/.
[3] http://medlineplus.gov/.
[4] http://www.ncbi.nlm.nih.gov/Taxonomy/.
[5] http://www.ncbi.nlm.nih.gov/pubmed/.
[6] http://www.novoseek.com/Welcome.action.
[7] http://www.informatics.jax.org/.

different machine learning classifiers such as Support Vector Machines (SVN) or Naïve Bayes were used by a variety of teams [26–29]. Regarding the representation of documents, several techniques such as porter stemmer algorithm, selection of n-grams, and stop-words were used, achieving the best results in those approaches that involved the use of MeSH terms. However, the best results only achieved 0.66 of *F*-measure.

Closer to our goals, the extraction of protein–protein interactions (PPI) from texts is one of three tracks proposed by BioCreAtIvE Challenges to tackle the problem of classification of articles from PubMed abstracts for database curation relevant to protein–protein interactions. A detail description of the subtasks as well as a comprehensive review of the participating systems can be found in [30,31]. Most participants used machine learning tech-niques such as SVM, Naïve Bayes or Maximum Entropy classifiers. Regarding the representation of the documents, participating teams mostly used the traditional bag-of-words approach with small variations. Stemming, POS tagging, Biomedical Named Entity Recognition or integration of knowledge from biological resources were the most used strategies to build the feature vector. In the BioCreative II Challenge, the training corpus consisted of 3536 PPI-relevant (positive) abstracts and 1959 non-relevant (negative) abstracts. The system presented in [32] achieved the best perfor-mance with a precision of 0.71, and a recall of 0.87. This approach used an SVM classifier and applied the abovementioned prepro-cessing techniques for adequate document representation. In addi-tion, more sophisticated methods such as abbreviation resolution were also introduced. In the last challenge, BioCreative II.5, the cor-pora for the evaluation consisted of 1190 full articles from FEBS Letters.[8] The best system [33] was a Naïve Bayes classifier imple-mented using citation features such as cited PMIDs (unique number assigned to each PubMed citation) and citation authors. The classifier achieved an *F*-measure of 0.63, a precision of 0.57 and a recall of 0.70, lower than the best ones in the previous challenge. This decline in performance may be due to the classification of full articles, which involves greater complexity than abstracts.

In the pharmaceutical domain, Duda et al. [34] used an SVM classifier to identify drug–drug interactions articles. The authors manually built a corpus composed of 2000 MedLine abstracts (1800 negatives and 200 positives). Two different document representations were used: the former is based on the use of UMLS identifier concepts generated by MMTx, and the latter is based on the common bag-of-words model, but MeSH terms are also included. The results showed that the second representation achieved better performance (0.99 of AUC) than the approach based on CUIs (0.98 of AUC).

In short, most approaches for biomedical text classification use machine learning methods such as SVM or Naïve Bayes. Regarding the document representation, the approaches range from the common (binary, TF or TF-IDF) bag-of-words model to the use of more sophisticated Natural Language Processing (NLP) techniques such as chunking or biomedical named entity recognition. Semantic information from biomedical resources has also been tentatively used [35]. While most approaches achieve a high recall, there is a need for further improvement in precision (which does not exceed 71%). Classification tasks are mainly linked to curate biological databases, simulating the task of curators for genomic databases (like MGI or FlyBase [36]) or protein interaction databases (such as IntAct [37] or MINT [38]). However, few approaches have tackled the classification of documents related to the pharmaceutical re-search domain.

In this paper, a comprehensive study of several machine learning algorithms is addressed in order to determine which algorithm is the most suited for drug target article identification task. As this is the first work that addresses this issue, a corpus has been created in order to fairly evaluate and compare the algorithms.

3. Our proposal

The main goal of our proposal is to maintain a service that queries PubMed in a methodical and automated manner. Each new article in MedLine can be classified as drug target or not, and sent to drug target databases, which can update their data adequately.

The development of this system needs to address two problems: the construction of a corpus for drug target article classification, which is not yet available, and the learning of patterns from the corpus for classification purposes. The description of the corpus, its construction and the techniques explored for classification are described in the following subsections.

3.1. Building the corpus

We have built a corpus of positive and negative drug target abstracts from DrugBank and PubMed. The corpus was created with abstracts published between 1995 and 2001. About 5% of all articles in MedLine concern drug targets. Such distribution was measured querying PubMed about abstracts with the UMLS synonyms of the term "biological target". In this way, an article was marked as related to drug target if it contained (or was annotated in MedLine with) at least one of these synonyms. A set of 4365 abstracts (1500 of them referred to drug target) was collected. Positive examples were randomly selected from the references in DrugBank which were recovered with the help of the RobotMaker[9] tool. Negative examples were randomly selected among MedLine abstracts which were not marked as drug target articles. Both sets contain only abstracts in the time range 1995–2001, and the distribution amongst drug target and no drug target abstracts observed in MedLine for each year was maintained.

In order to assess the quality of the negative examples set, a 5% (143) sample was randomly selected and manually evaluated with the help of a pharmacist. The evaluation showed that none of the abstracts were related to drug targets, supporting the quality of the corpus.

3.2. Preprocessing the corpus

A general schema of the corpus preprocessing appears in Fig. 1. The dotted squares are the final recovered data. After the set of randomly selected abstract examples has been recovered from MedLine and DrugBank (as explained in the previous section), a set of features are extracted in order to build a representation of each article. We were able to obtain, querying PubMed, title, ab-stract and MeSH and chemical concepts associated to each ab-stract, because they are fields of MedLine database.

Chemical concepts were extracted using *NameOfSubstance* data in *chemical list* field at MedLine database, which belong to MeSH vocabulary. Therefore, we define two features: *chemical concepts* with the content of *chemical list field* in MedLine, and *MeSH* feature with the non-chemical concepts at MeSH field. These two features are used in the training set to express the appearance of the related concept with the corresponding example.

From title and abstracts we recovered the semantic types and groups, as well as the stemmed words and drug families associated to each of these parts. Word stems have been extracted using the Porter stemmer algorithm.

[8] http://www.febsletters.org/, split evenly into training and test set.

[9] http://openkapow.com/.

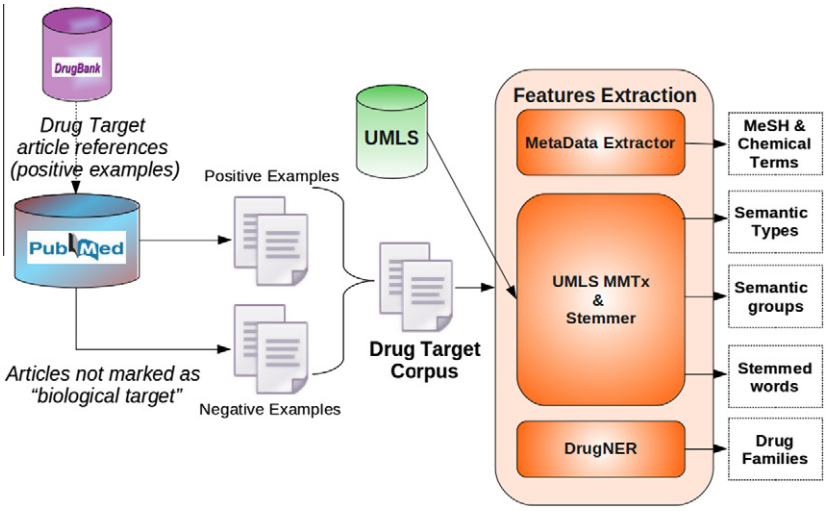


Fig. 1. Corpus preprocessing.

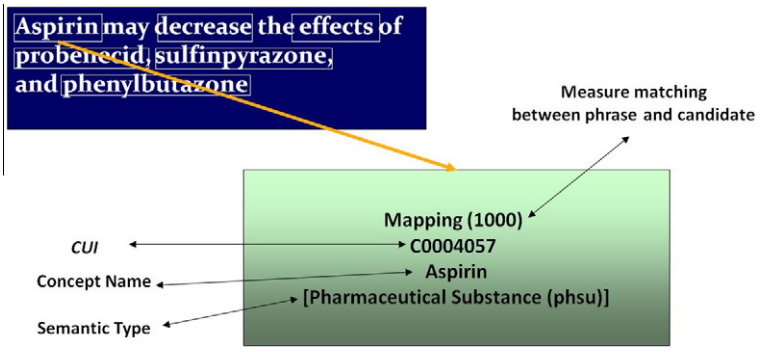


Fig. 2. Concept 'Aspirin' retrieved by MMTx.

3.2.1. MMTx processing

MMTx [19] analyzes the text syntactically in order to split it into components of different syntactic levels: sentences, phrases, lexical elements and tokens. Then, MMTx generates variants from each phrase to look up the concepts in the UMLS Metathesaurus that contain one or more of these variants. In this way, a set of candidate concepts are retrieved from the UMLS Metathesaurus and are evaluated against the phrases using a linguistically rigorous metric. Those candidates that best fit the text are selected and organized into a final mapping. Furthermore, MMTx also retrieves the semantic types assigned to each concept. Thus, each phrase may be related to one or more UMLS concepts together with their semantic types.

Fig. 2 shows what information is retrieved by MMTx for the phrase *"Aspirin may decrease the effects of probenecid, sulfinpyrazone, and phenylbutazone*". For this phrase the final mapping of MMTx consists of an unique concept, *Aspirin*, with identifier (CUI) *C0004057* and semantic type *Pharmacological substance*.

Semantic types offer very useful information. However, it would be helpful if the semantic annotation of titles or abstracts had less granularity. There are 136 semantic types, grouped in 15 semantic groups [39] in the UMLS Semantic Network. For example, "Anatomy" (ANAT) semantic group refers to concepts associated, amongst others, to "Anatomical Structures" and "Tissue" semantic types. Therefore, we used the links between semantic type and groups in UMLS Semantic Network to recover the semantic groups associated to titles and abstracts in our examples.

3.2.2. DrugNer

Each abstract is preprocessed by the DrugNer [40] system for drug name recognition and classification. DrugNer extends the information provided by MMTx, by the use of the nomenclature rules recommended by the WHO International Nonproprietary Names (INNs) Program[10] to identify and classify pharmaceutical substances. Once abstracts have been processed by MMTx and the phrases occurring in the text are annotated and related to concepts of the UMLS Metathesaurus and to semantic types of the UMLS Semantic Network, a second rule-based module classifies the pharmacological substances (that is, those phrases that have been related to the UMLS semantic types which represent generic drugs: "antb" or "phsu") occurring in texts into pharmacological families. This module implements the naming convention rules defined by the WHOINNs Program to facilitate the identification and classification of pharmaceutical substances or active ingredients. The rules are based on the common affixes selected and defined by WHOINN. These common affixes aid healthcare professionals to recognize that the substance belongs to a group of substances having similar pharmacological activity or chemical structure.

Table 1 shows some of the affixes used in the classification of drug names. The full list and the affix classification can be found in [41].

[10] http://www.who.int/medicines/services/inn/en/.

Table 1
Some affixes recommended by WHOINN.

Affixes	Drug family
-flurane	General anaesthetics, volatile
-arol, -grel-, -irudin, -pafant, -troban	Anticoagulants
-oxetine	Antidepressants
-afil, -dil, -entan	Vasodilators

Table 2
Examples of matching phrases and affixes.

Drug	Suitable affixes	Most suitable affix
Azelnidipine	-dipine, -pine, -ine, -ni-	-dipine
Lopinavir	-navir, -vir-	-navir
Amiodarone	-arone, -one, -io-,	-arone
Minocycline	-cycline, -ine	-cycline
Aripiprazole	-piprazole, -prazole	-piprazole

DrugNer scans the list of affixes in order to build the suitable regular expression for each affix. For example, for the afix -adol-, the regular expression should be [A-Za-z0-9]*adol[A-Za-z0-9]*. Therefore, any alphanumeric string which contains the afix -adol-is recognized by this regular expression. Once the regular expressions have been built, the module tries to match the text of each phrase with the regular expressions in order to detect the possible affixes, which may classify the phrase. In the case in which several regular expressions can be matched with the text of the phrase, the module selects the longest affix.

Table 2 shows some examples. When a correct affix is found, the pharmacological or chemical family associated with the affix is added to the phrase. The rules are not only applied to the phrases that have been classified as pharmacological substances or as antibiotics by the MMTx program, but also to those for which MMTx did not found any candidate concept in UMLS. Thus, these phrases are possible new candidates for drug names that are not included in UMLS Metathesaurus.

A more detailed description of the DrugNer system is described in [40]. A corpus of 875 MedLine abstracts was automatically annotated by DrugNer, and subsequently manually-evaluated by a pharmacological expert. This corpus is available for research purposes[11], but unfortunately, it contains some syntactic and semantic errors made by the MMTx program, but we have not addressed this problem yet.

3.3. Document representation

All features previously described are used to construct the final dataset for drug target article classification. The set of collected features are summarized as follows:

1. Chemical terms (*chem*): UMLS terms about drugs and chemical products used by the authors to characterize their article (extracted from the field MESH of PubMed database),
2. MeSH terms (*MeSH*): other UMLS terms, different from the chemical terms, used by the authors to characterize their article (extracted from the field MESH of PubMed database),
3. The stemmed words of the title (*stemTitle*),
4. The stemmed words of the abstract (*stemAbstract*),
5. Drug affixes (*drug*): the drug families mentioned in the abstract (extracted by using DrugNer system),
6. Semantic types and groups (*semTypeGroup*): semantic types and groups of the mentioned UMLS terms (extracted by using MMTx and Semantic Network).

[11] http://basesdatos.uc3m.es/index.php?id=359.

The first two features are represented as boolean vectors, describing whether chemical and MeSH terms appear in the respective PubMed data of the article. Title and abstract features are transformed using the classical string feature representations: term frequency (TF), term frequency-inverse document frequency (TF-IDF) and term frequency-inverse document frequency with normalization (TF-IDF-Norm). We analyze the effect of using each kind of representation in the classification results. All other features are integer data, describing the frequency with which a concept appears in the respective article. The notation used in figures and tables in the remainder of the paper is specified in the above list in italic. Affixes *TF*, *TF-IDF* and *TF-IDF-Norm* are used to clarify which kind of string representation is used; *Title* and *Abstract* affixes are used to specify the context in which a determined feature is extracted, and *AllVars* is the notation used when all features are considered.

3.4. Machine learning techniques

A set of machine learning algorithms for binary drug target article classification have been tested: C4.5 [42]; Bayesian statistics as Naïve Bayes [43], Complement Naïve Bayes [44] (CNB), Bayes Network [45] and DMNBtext [46]; LogitBoost [47] and its combination with trees, the Logistic Model Trees (LMT) [48,49]; Fuzzy Lattice Reasoning (FLR) [50,51]; Support Vector Machine (SVM) [52], and HyperPipes [53] (HP).

These algorithms cover different kinds of machine learning techniques (decision trees, Bayesian statistics, feature space division, etc.) and share characteristics that make them interesting to our analysis: (a) they all have been used in text classification tasks with good results; (b) they have efficient implementations; and (c) the resulting model allows a fast classification processing. All experiments have been performed according to the classical schema for selecting optimal classification parameters, i.e, first, we have selected attributes in order to eliminate dependent sets of features and then, we have optimized the parameters for each clas-sifier. We finally compare of the results and select the best param-eter configurations.

4. Experimental results

Several experiments were carried out in order to validate the proposed classifier for drug–target articles. Since the observed ratio between the number of positive and negative examples is highly unbalanced, we have studied the effect of using different proportions in positive and negative examples in the training set. Therefore, we have considered 4 training datasets containing 5% (real distribution), 10%, 20% and 50% of positive examples respectively, in which the different training sets share as many examples as possible. This solution reduces the possibility of meaningless results due to differences in training data. In Fig. 3 the four training datasets are represented with different colors, to show the proportion and overlap between their positive and negative subsets.

All experiments were performed using the Weka package [53], and a 10-fold cross-validation framework was employed for testing the results. A parameter selection process was performed for each training set. An exhaustive search was performed for those algorithms with more than two parameters, a grid search for those algorithms with two parameters, and the optimizing tools provided for libSVM package [54] were used in the case of the SVM classifier.

In order to evaluate the classification results we have computed the ROC area measure, because in the last years many authors have recognized its importance in order to give a more realistic vision of the quality of binary classifications [56]. This measure gives an idea

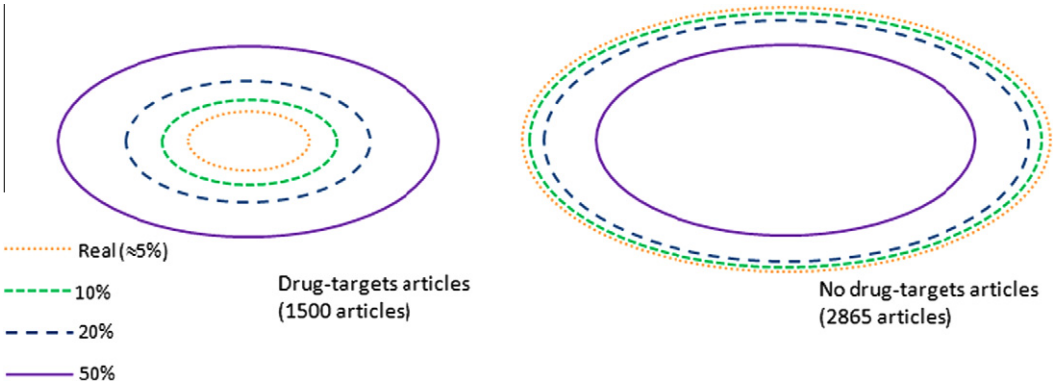


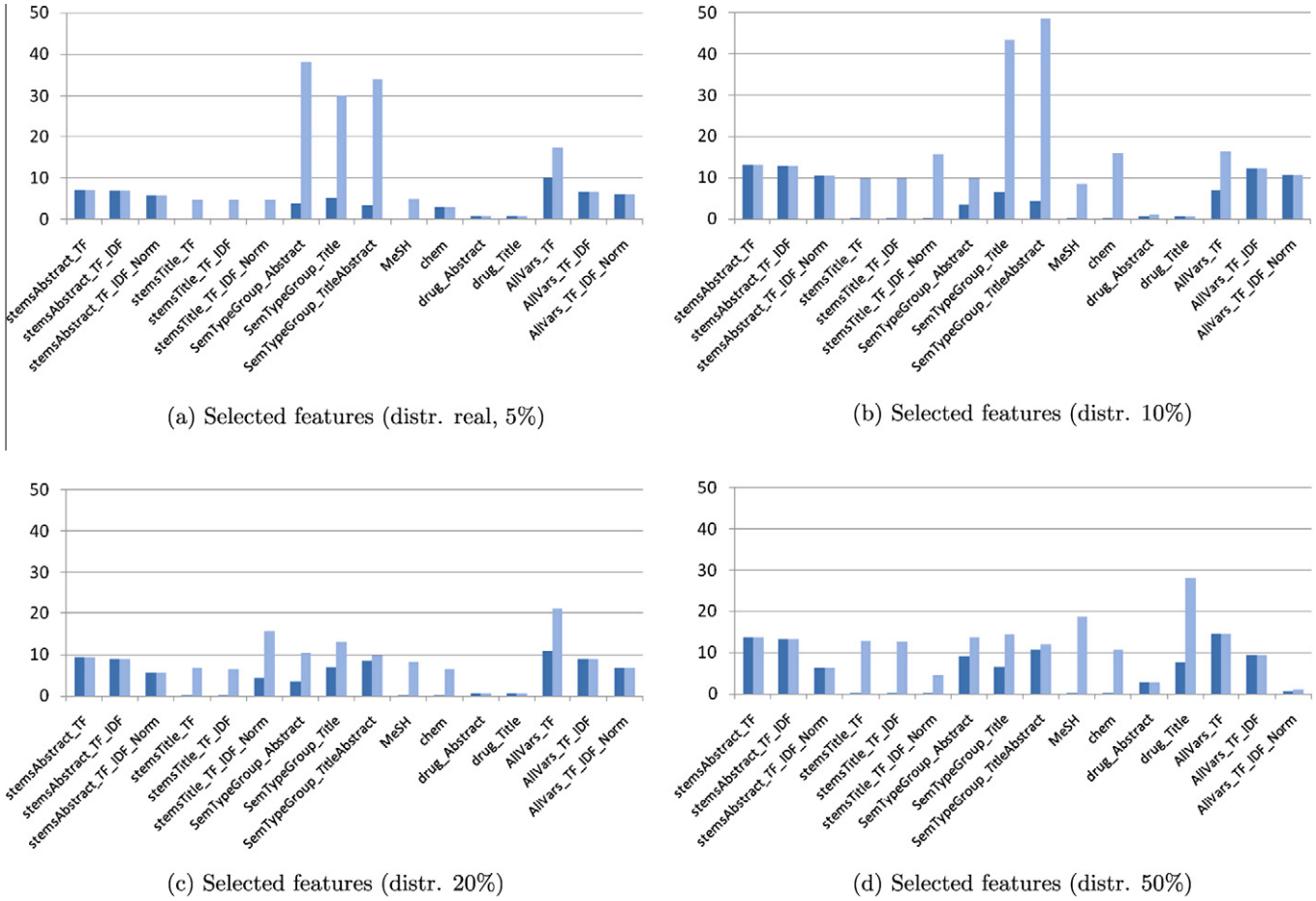
Fig. 3. Distribution of positive and negative examples in the four analyzed training sets.

of how close the predictions of a classifier are to the ideal ratio (1.0) between true and false positive rates. However, ROC area measure offers an excessively optimistic assessment of the results when there is a large skew in the class distribution [57]. For this reason, we have also employed the classical metrics of precision (P), recall (R) and F_β-measure, more suitable to tasks with a large skew in the class distribution. Precision is associated with the capacity of classifying instances correctly, while recall is associated with the capacity of classifying as many instances as possible; the F_β-measure offers a global description considering both precision and recall. For F_β-measure we have used the parameter $\beta = \{1, 2\}$: $F_\beta = (1 + \beta^2)P^*R/(\beta^{2^*}P + R)$. With $\beta = 1$, the classical F_1-measure is obtained; when $\beta = 2$, an overall performance is obtained which gives more importance to recall.

4.1. Feature selection

The feature selection phase, also known as attribute selection, variable selection or feature reduction is used in Machine Learning for selecting a subset of relevant features in order to construct robust models from datasets. For feature selection, Correlation Feature Subset Selection (CFS) algorithm [59], Symmetrical Uncertain (SymUncert) [60], Information Gain (InfoGain) [60], Gain Ratio (GainRatio) [60], Relief [65,66] and Chi Squared (ChiSquared) [67] metrics have been used in this work.

Fig. 4 represents the minimum and maximum percentages of dimensionality after reduction for each feature and training set distribution. Very similar performances are obtained for the training sets with 20% or 50% of positive examples (Figs. 5(a) and (b)).



(a) Selected features (distr. real, 5%)

(b) Selected features (distr. 10%)

(c) Selected features (distr. 20%)

(d) Selected features (distr. 50%)

Fig. 4. Minimum (left columns) and maximum (right columns) percentages of features selected per distribution.

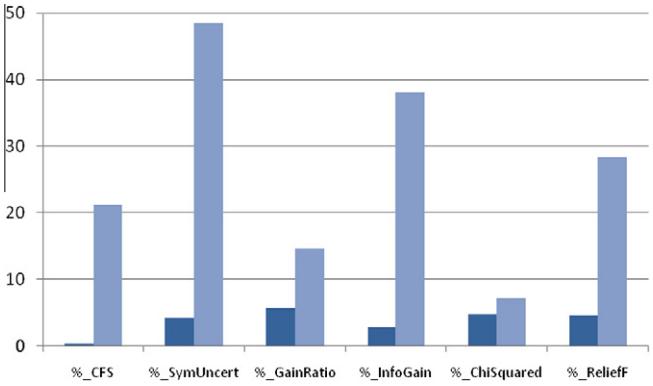


Fig. 5. Minimum and maximum percentage of feature selected per algorithm.

When the set of positive examples is 5% or 10%, semantic types and semantic groups cannot be reduced very much, but this does not affect the reduction when all features are used together (see the last three columns of Figs. 4(a) and (b)).

Comparing the four histograms of Fig. 4, the general behavior of dimensionality reduction for all features can be analyzed. Features *stemTitle* and *stemAbstract* features obtain drastic reductions of at least 82% (and up to over 99.9%). All semantics features (semantic types and groups, MeSH and chemical terms as well as drugs) are drastically reduced in the majority of the cases between 85% and 95%. These data confirm that ontologies describing semantic types and groups, UMLS concepts and pharmacological families define orthogonical spaces of knowledge that can be useful for clustering and classification tasks.

However, the severe reduction of the drug family feature indicates the high dependence among drug families (the reference to one of them implies a subsequent reference to others). This may explain why the use of such feature does not help in drug–target article classification, as we show in Section 4.3.

When all features are used together, selection reduction is between 72% and 96%, a range that can be explained considering the orthogonality of the majority of features previously described, and the reductions obtained in each case.

Fig. 5 shows the results obtained by each feature selection algorithm in the process of feature dimensionality reduction. The Chi-squared measure gets the highest reduction, but just for the corpus associated to the real data distribution (in the other cases the reduction is at most 50%). The utility of a feature selection measure is linked its classification accuracy. The above analysis on feature reduction can be especially useful when the number of features should be reduced for computational efficiency while running data mining algorithms. In such cases, we recommend to test the performances with Gain Ratio, CFS selections and/or ReliefF selections, since the Symmetrical Uncertain and Information Gain show very low reductions considering different features and distributions.

4.2. Analyzing string feature representations

We have analyzed three different representations for string features: term frequency (TF), term frequency-inverse document frequency (TF-IDF) and term frequency-inverse document frequency with normalization (TF-IDF-Norm). For this analysis we have created 36 corpora from the original training data which are the combination of: (a) each positive set distribution; (b) each of the three specific string representations; and (c) each of the following set of features: (c.1) all features, (c.2) just the *stemTitle* features or (c.3) just the *stemAbstract* features. All these corpora have been used to train the algorithms presented in Section 3.4.

Table 3 shows the best classification results for each combination of training data and Table 4 shows the classifiers that have achieved such results. Each classifier is described by its selection feature and classification algorithms. We verified by Mc Nemar hypothesis test [63] that no other classifiers constructed with the same positive example distribution are statistically equivalent to the best ones appearing in the table.

All metrics achieve the best results using the same configuration (feature and string representation type) for each distribution (see values marked with a star in Table 3). For a 5% distribution of positive examples, the ideal configuration is *stemAbstract* with *TF* representation; for 10% and 50% distributions, the best solution is to use *AllVars* features with *TF-IDF* representation; for 20% distribution, *AllVars* features with *TF-IDF-Norm* representation. The exceptions to this pattern (F_2 measure for 50% distribution and *ROC* measure for 20%) improve only by a 0.01 their corresponding "ideal" configurations.

The corpus composed of the *stemAbstract* features shows better performance than the corpus composed of the *stemTitle* features, for all metrics, especially when normalization is performed (see rows *TF-IDF-Norm* in Table 3). The reason for this difference could be that the normalization of *TF-IDF* values with respect to document length gives more importance to words belonging to short abstracts (or titles). In this way, essential patterns for the classification task in large texts may be not detected. *TF* and *TF-IDF* representations show similar results for both *stemAbstract* and *stemTitle* features. For real and 10% distributions, *TF* representation obtains much better F_1 and F_2 scores for these features than using *AllVars* features. This result corresponds with previous works based only on bag of words for the classification task. For all distributions (except 5%), all features with a *TF-IDF* (instead of TF) representation improve by at least a 3% the results achieved by the *stemTitle* and

Table 3
Classification results using different string feature representation.

String repres.	Feature	Real (5%)	10%	20%	50%
F_1					
TF	stemTitle	0.871	0.918	0.800	0.916
	stemAbstract	0.893*	0.928	0.817	0.923
	AllVars	0.641	0.736	0.832	0.917
TF-IDF	stemTitle	0.875	0.918	0.779	0.920
	stemAbstract	0.887	0.928	0.824	0.924
	AllVars	0.866	0.938*	0.871	0.949*
TF-IDF-Norm	stemTitle	0.669	0.662	0.801	0.890
	stemAbstract	0.634	0.843	0.830	0.923
	AllVars	0.634	0.854	0.880*	0.936
F_2					
TF	stemTitle	0.865	0.893	0.866	0.947
	stemAbstract	0.904*	0.925	0.878	0.950
	AllVars	0.669	0.793	0.873	0.941
TF-IDF	stemTitle	0.871	0.895	0.871	0.950
	stemAbstract	0.902	0.925	0.891	0.957*
	AllVars	0.897	0.953*	0.856	0.956
TF-IDF-Norm	stemTitle	0.754	0.776	0.824	0.927
	stemAbstract	0.795	0.911	0.889	0.952
	AllVars	0.743	0.924	0.897*	0.948
ROC					
TF	stemTitle	0.951	0.944	0.953	0.960
	stemAbstract	0.969	0.961	0.971	0.976
	AllVars	0.962	0.973	0.970	0.966
TF-IDF	stemTitle	0.950	0.939	0.959	0.959
	stemAbstract	0.970*	0.961	0.970	0.976
	AllVars	0.955	0.977*	0.980*	0.988*
TF-IDF-Norm	stemTitle	0.908	0.924	0.944	0.949
	stemAbstract	0.967	0.964	0.966	0.970
	AllVars	0.955	0.971	0.971	0.975

Table 4
Classifiers associated to classification results in Table 3.

String repres.	Feature	Real (5%)	10%	20%	50%
F_1					
TF	stemTitle	ChiSquared;FLR*	InfoGain;FLR*	SymUncert;SVM	SymUncert;CNB*
	stemAbstract	SymUncert;FLR*	SymUncert;FLR*	SymUncert;DMNBtext	SymUncert;DMNBtext
	AllVars	CFS;SVM	CFS;LogitBoost	CFS;SVM	CFS;BayesNet*
TF-IDF	stemTitle	InfoGain;FLR*	GainRatio;FLR*	GainRatio;SVM	InfoGain;CNB*
	stemAbstract	InfoGain;FLR*	GainRatio;FLR*	GainRatio;DMNBtext	InfoGain;CNB*
	AllVars	GainRatio;FLR*	GainRatio;FLR*	GainRatio;FLR*	GainRatio;FLR*
TF-IDF-Norm	stemTitle	InfoGain;FLR*	CFS;CNB*	SymUncert;SVM	InfoGain;CNB*
	stemAbstract	InfoGain;CNB*	GainRatio;FLR*	GainRatio;SVM	InfoGain;DMNBtext*
	AllVars	GainRatio;SVM*	GainRatio;FLR*	GainRatio;SVM	GainRatio;SVM*
F_2					
TF	stemTitle	ChiSquared;FLR	InfoGain;FLR	SymUncert;CNB	SymUncert;CNB
	stemAbstract	SymUncert;FLR	SymUncert;FLR	SymUncert;CNB	SymUncert;CNB
	AllVars	CFS;BayesNet	CFS;BayesNet	CFS;NaiveBayes	CFS;BayesNet
TF-IDF	stemTitle	InfoGain;FLR	GainRatio;FLR	GainRatio;CNB	InfoGain;CNB
	stemAbstract	InfoGain;FLR	GainRatio;FLR	GainRatio;CNB	InfoGain;CNB
	AllVars	GainRatio;FLR	GainRatio;FLR	GainRatio;FLR	GainRatio;FLR
TF-IDF-Norm	stemTitle	InfoGain;FLR	CFS;CNB	SymUncert;CNB	InfoGain;CNB
	stemAbstract	InfoGain;CNB	GainRatio;FLR	GainRatio;CNB	InfoGain;CNB
	AllVars	GainRatio;FLR	GainRatio;FLR	GainRatio;FLR	GainRatio;SVM
ROC					
TF	stemTitle	ChiSquared;NaiveBayes	InfoGain;CNB	SymUncert;DMNBtext	SymUncert;DMNBtext
	stemAbstract	SymUncert;NaiveBayes	SymUncert;BayesNet	SymUncert;HP	SymUncert;HP
	AllVars	CFS;NaiveBayes	CFS;LogitBoost	CFS;LogitBoost	CFS;LogitBoost
TF-IDF	stemTitle	InfoGain;NaiveBayes	GainRatio;NaiveBayes	GainRatio;DMNBtext	InfoGain;DMNBtext
	stemAbstract	InfoGain;NaiveBayes	GainRatio;BayesNet	GainRatio;HP	InfoGain;HP
	AllVars	GainRatio;FLR	GainRatio;FLR	GainRatio;HP	GainRatio;HP
TF-IDF-Norm	stemTitle	InfoGain;NaiveBayes	ReliefF;LogitBoost	SymUncert;LogitBoost	InfoGain;DMNBtext
	stemAbstract	InfoGain;BayesNet	GainRatio;FLR	GainRatio;DMNBtext	InfoGain;DMNBtext
	AllVars	GainRatio;BayesNet	GainRatio;FLR	GainRatio;DMNBtext	GainRatio;HP

stemAbstract features. The ROC Area shows very high (optimistic) values for all distributions and representations, obtaining the maximum values when *TF-IDF* representation is used. Taking into account these insights, we justify the preference to use the *TF-IDF* representation for the string features, and the results showed in the next sections are thus based on the use of the *TF-IDF* representation.

With respect to the classifiers associated to each result (see Table 4), the following issues can be drawn. In the majority of cases, the best F_1 and F_2 values are achieved using the same combination of algorithms. A prevalence of the combination *InfoGain* or *GainRatio* with the FLR classifier can be observed when there are few positive examples (5% or 10%), whilst *GainRation* with *SVM* or *CNB* prevail for the other distributions. The best results in ROC area are obtained with probabilistic approaches, such as BayesNet and NaiveBayes (5% and 10% distributions), as well as with text-directed approaches such as HyperPipes and DMNBtext.

4.3. Feature analysis

We have studied the behavior of the features for the different positive set distributions in the classification task (see Table 5). We have classified the features into five groups, according to the type of information that they represent: (1) *stemTitle* and *stemAb-stract* features, (2) MeSH and chemical terms, (3) semantic types and groups in titles and abstracts, (4) drug affixes in title and ab-stracts and (5) all features.

The features of the first group show a similar behavior, and the use of abstracts is advantageous in most cases for all measures and distributions, with up to a 5% improvement. This result is easily justifiable by the relative increase of knowledge offered by the abstract in relation to the article's title only.

A somewhat unexpected result is obtained for the second group of features: *MeSH* terms are less informative than the chemical terms for real and 10% distributions (up to 5% of difference). A contrary situation is observed for 20% and 50% distributions (up to 17% of difference). In the case of ROC area, the *MeSH* terms are more discriminative than *chem* ones, except for the real distribution.

The three features of the family *semTypeGroup* show very similar results between them, with a difference of less than 2% in most of the cases. The classification performance improves slightly when *semTypeGroup_TitleAbstract* is used (except for F_2 and ROC measures in the case of 20% of positive examples), but at the price of the additional effort of analyzing and using the semantic information contained in abstracts. In contrast to stems, semantic types and groups of titles provide better classification results than semantic types and groups of abstracts.

The fourth group shows an unusual behavior compared to the rest of the features. In fact, the drug families mentioned in title and abstract of articles are not useful in the classification process. The only acceptable score is achieved for F_2 measure when the dataset with 50% of positive examples is used.

When all informative features are used, classification results are clearly better for all measures and positive class distribution equal or over 10% (in Table 5 the highest values per measure and distribution are marked with a star). Therefore, all above features give a contribution to the overall results.

The algorithms associated to the above results are shown in Table 6, in which we have omitted the rows associated to drug families because these attributes are not useful for our classification task. We verified by Mc Nemar hypothesis test that no other classifiers constructed with the same positive example distribution are statistically equivalent to the best ones appearing in the table. For stems (*stemTitle* and *stemAbstract*), *MeSH* and *chem* features,

Table 5
Classification results by feature and distribution of positive examples.

Feature	Real (5%)	10%	20%	50%
F_1				
stemTitle	0.875*	0.918	0.779	0.920
stemAbstract	0.887	0.928	0.824	0.924
MeSH	0.813	0.848	0.829	0.930
chem	0.856	0.886	0.716	0.859
semTypeGroup_Title	0.459	0.620	0.701	0.873
semTypeGroup_Abstract	0.422	0.563	0.697	0.873
semTypeGroup_TitleAbstract	0.492	0.635	0.740	0.887
drug_Title	0.000	0.013	0.201	0.673
drug_Abstract	0.105	0.081	0.207	0.664
AllVars	0.866	0.938*	0.871*	0.949*
F_2				
stemTitle	0.871	0.895	0.871	0.950
stemAbstract	0.902*	0.925	0.891*	0.957
MeSH	0.770	0.795	0.871	0.944
chem	0.823	0.843	0.704	0.849
semTypeGroup_Title	0.557	0.692	0.784	0.903
semTypeGroup_Abstract	0.510	0.663	0.769	0.902
semTypeGroup_TitleAbstract	0.561	0.700	0.712	0.904
drug_Title	0.000	0.008	0.200	0.836
drug_Abstract	0.071	0.053	0.208	0.827
AllVars	0.897	0.953*	0.856	0.956*
ROC				
stemTitle	0.932	0.938	0.959	0.959
stemAbstract	0.953	0.958	0.970	0.976
MeSH	0.870	0.962	0.966	0.968
chem	0.900	0.907	0.817	0.867
semTypeGroup_Title	0.936	0.938	0.934	0.926
semTypeGroup_Abstract	0.936	0.923	0.926	0.918
semTypeGroup_TitleAbstract	0.943	0.942	0825	0.933
drug_Title	0.500	0.503	0.504	0.516
drug_Abstract	0.521	0.532	0.518	0.516
AllVars	0.955*	0.977*	0.980*	0.988*

the algorithm obtaining the best results is a combination of *InfoGain* or *GainRatio* feature selection algorithms with the FLR classification algorithm. For semantic type and groups features, it is not clear what configuration allows to obtain the best results. However, *CFS* with *BayesNet* as well as *InfoGain* with *DMNBtext* are the most frequent combinations. When all features are used, the FLR algorithm (or HyperPipes in the case of ROC area, for 20%and 50% of positive examples), preceded by a Gain Ratio feature selection, achieved the best results.

Comparing the results of using different distributions of positive and negative examples, we observe that a 20% of positive examples does not guarantee to obtain higher results than with a 10% of positive examples. The use of all features shows increasing F_1, F_2 scores as the distribution of positive examples is increased, but with lower values for the 20% distribution. This observation fits with various unbalanced biomedical binary classification tasks, in which the distribution is adjusted to 10% independently from the real distribution of the classes, like in [34].

A detailed analysis of the above results allows us to determine the following orders, representing the relative importance of the features for classification:

- For the distributions of 5% and 10% of positive examples: (1) *stemAbstract* features, (2) *stemTitle* features, (3) MeSH and chemical terms, (4) semantic type and groups features, and (5) drug families features.
- For the distributions of 20% and 50%: (1) MeSH terms, (2) *stemAbstract* features, (3) *stemTitle* features, (4) semantic types and groups features, (5) chemical terms, and (6) drug families features.

In addition, we have performed a detailed analysis of the results to choose the most informative features for each of the classifiers. Table 7 shows the most informative features of the trained classifier models. The features have been selected taking into account the ROC area as well as the F_1 and F_2 scores. We can observe that most algorithms benefit from using all features to train their models.

4.4. Best classifier configurations

All configurations providing the best result for at least one measure (precision, recall, F_1, F_2 or ROC area) have been included in the set of best classifiers, independently of the class distribution. Table 8 shows the best configurations and their scores. Each configuration is specified by an identifier (first column) described by: (a) the class instances distribution (real (R), 10, 20, 50); (b) the used feature(s); (c) the measure for feature selection; and (d) the statistical machine learning algorithm employed. For example, *50;StemAbstract;InfoGain;FLR* means that the 50% distribution of positive examples was used, the set of features consists of the *stemAbstract* features which are filtered using the *InfoGain* measure, and the classification is performed using the FLR classifier. Only 10% and 50% distributions are represented in the set of the best configurations. The last two rows of Table 8 show the best scores for the real and 20% distributions. When the positive class represents 5% or 20%, all measures are relatively low, except for the ROC area.

According to the F_1 measure, which gives the same importance to precision and recall, the best classifier is *50;AllVars;GainRatio;FLR* obtaining high quality values for all measures, with a 0.95 of F_1-measure, 0.96 of F_2-measure and 0.95 of ROC Area. Classifier *10;AllVars;GainRatio;FLR* achieves similar results, and both classifiers share the same configuration, except the distribution of positive examples. We believe that the best classifier is thus the first in Table 8, because it obtains results similar to those obtained by other classifiers, but needs less positive examples to train its model. We used Mc Nemar's test to examine if the *10;AllVars;GainRatio;FLR* classifier is significantly better than the other classifiers. The null hypothesis H0 is *no preference towards the 10;AllVars;GainRatio;FLR classifier*. The alternative hypothesis H1 is defined as *there is a preference towards the 10;AllVars;GainRatio;FLR classifier*. We use a 95% confidence level for verifying/falsifying the hypothesis. The test results (see Table 9) indicate that, (in the 95% of the cases) the *10;AllVars;GainRatio;FLR* classifier obtains results equal to the classifier *50;AllVars;GainRatio;FLR* (which have the same configuration but a different class example distribution), and is significantly better than the other classifiers.

The FLR classifier divides the parameter space in lattices, in which abstracts sharing a common subset of properties and having some similarities are grouped. The classifier works with fuzzy intervals instead of fuzzy numbers. This allows to produce a reduced set of fuzzy rules which achieves a clear and simple knowledge representation of the drug target abstracts. The FLR classifier has been used for addressing several classification tasks such as ambient air quality assessment [61] and ocean satellite image recognition [62]. Its effectiveness has been showed by the high precision and recall values obtained in comparison with other classifiers, such as C4.5, in which the number of rules generated is often excessive.

The HyperPipes classifier considers the ranges observed in the training data for each feature and class. Then, the classifier uses this information to select the class that contains the largest

Table 6
Algorithms associated to classification results in Table 5.

Feature real	(5%)	10%	20%	50%
F_1				
stemTitle	InfoGain;FLR	GainRatio;FLR	GainRatio;SVM	InfoGain;CNB
stemAbstract	InfoGain;FLR	GainRatio;FLR	GainRatio;DMNBtext	InfoGain;CNB
MeSH	InfoGain;FLR	SymUncert;FLR	SymUncert;SVM	SymUncert;SVM
chem	InfoGain;FLR	SymUncert;FLR	SymUncert;CNB	SymUncert;CNB
semTypeGroup_Title	InfoGain;DMNBtext	SymUncert;SVM	CFS;BayesNet	CFS;BayesNet
semTypeGroup_Abstract	InfoGain;DMNBtext	CFS;BayesNet	CFS;SVM	CFS;SVM
semTypeGroup_TitleAbstract	InfoGain;DMNBtext	SymUncert;DMNBtext	FilteredSubsetEval;SVM	CFS;SVM
AllVars	GainRatio;FLR	GainRatio;FLR	GainRatio;FLR	GainRatio;FLR
F_2				
stemTitle	InfoGain;FLR	GainRatio;FLR	GainRatio;CNB	InfoGain;CNB
stemAbstract	InfoGain;FLR	GainRatio;FLR	GainRatio;CNB	InfoGain;CNB
MeSH	InfoGain;FLR	SymUncert;FLR	SymUncert;NaiveBayes	SymUncert;FLR
chem	InfoGain;FLR	SymUncert;FLR	SymUncert;CNB	SymUncert;CNB
semTypeGroup_Title	InfoGain;BayesNet	SymUncert;BayesNet	CFS;BayesNet	CFS;BayesNet
semTypeGroup_Abstract	CFS;NaiveBayes	CFS;CNB	CFS;CNB	CFS;SVM
semTypeGroup_TitleAbstract	CFS;BayesNet	CFS;BayesNet	FilteredSubsetEval;SVM	CFS;SVM
AllVars	GainRatio;FLR	GainRatio;FLR	GainRatio;FLR	GainRatio;FLR
ROC				
stemTitle	InfoGain;FLR	GainRatio;FLR	GainRatio;DMNBtext	InfoGain;DMNBtext
stemAbstract	InfoGain;FLR	GainRatio;FLR	GainRatio;HP	InfoGain;HP
MeSH	InfoGain;FLR	CFS;NaiveBayes	SymUncert;LogitBoost	SymUncert;DMNBtext
chem	InfoGain;FLR	SymUncert;FLR	SymUncert;CNB	SymUncert;DMNBtext
semTypeGroup_Title	InfoGain;DMNBtext	CFS;BayesNet	CFS;BayesNet	CFS;BayesNet
semTypeGroup_Abstract	InfoGain;DMNBtext	CFS;BayesNet	CFS;BayesNet	CFS;DMNBtext
semTypeGroup_TitleAbstract	InfoGain;DMNBtext	SymUncert;DMNBtext	FilteredSubsetEval;SVM	CFS;DMNBtext
AllVars	GainRatio;FLR	GainRatio;FLR	GainRatio;HP	GainRatio;HP

Table 7
Most informative features for each classifier.

Classifier	Metrics		
	F_1	F_2	ROC
BayesNet	AllVars		
CNB	stemTitle (5%) stemAbstract (10, 50%); MeSH (20%)		stemTitle (5%); stemAbstract (10, 20, 50%)
DMNBtext	AllVars (5, 20, 50%); semTypeGroup (10%); stemAbstract (20%) AllVars (5, 50%);	AllVars (5, 50%); stemAbstract (10, 20%); MeSH (50%)	AllVars (5, 50%); stemAbstract (10, 20%);
FLR	stemAbstract (5%); semTypeGroup (10%); AllVars (20, 50%)		
HP	AllVars (5, 20%); semTypeGroup (10%) stemAbstract (50%)	AllVars (5, 20, 50%); semTypeGroup (10%);	
C4.5	AllVars (5%); semTypeGroup (10%); MeSH (20%); stemAbstract (50%)	AllVars (5%); semTypeGroup (10%); MeSH (20, 50%)	
LMT	MeSH (5, 20, 50%); semTypeGroup (10%)		
NaïveBayes	MeSH (5, 20%); semTypeGroup (10%); stemAbstract (50%)		
LogitBoost	stemAbstract (5, 50%); AllVars (10, 20%)	stemAbstract (5%); AllVars (10, 20, 50%)	
SVM	AllVars		

number of correct ranges for each test instance. This classifier has reported good results especially when a large number of features is considered, as in our case. Finally, the CNB classifier shows the lower scores amongst the best classifiers in Table 8, which may be due the assumption that features are independent, unrealistic in this domain.

It is difficult to compare our work to other approaches, because we are the first to address the problem of classification of drug target articles, and our experiments have been performed on a specific corpus for our task. Thus, our results are only partially comparable to other works. As mentioned in section 2.3, the corpus used in the BioCreative II Challenge has a higher proportion of positive abstracts (64.3%) than our corpus. However, the best performance in the challenge was only 0.78 for *F*-measure. Our results also improve those reported in the BioCreative II.5 Challenge (where the best *F*-measure was 0.63 [33]), although the classification task there was substantially more difficult, being applied to full articles. Many works on classification of protein interaction abstracts have used the SVM classifier, although they have not performed a comparative analysis among different classifiers to the depth and extent reported here. As reference, Table 10 shows the best results for the different positive example distributions when SVM is used. Increasing the number of positive examples allows to improve all measures.

5. Conclusions and future work

To the best of our knowledge, this is the first work considering the classification task for drug–target articles to aid drug–target database curation. In addition, our study provides a dataset which can serve as a benchmark for encouraging the development of new approaches.

Table 8
Best configurations for all distributions.

Id	P	R	F_1	F_2	ROC
10;AllVars;GainRatio;FLR	0.915	0.963	0.938	0.953	0.977
50;AllVars;GainRatio;HP	0.966	0.917	0.941	0.926	0.988
50;AllVars;GainRatio;FLR	0.936	0.961	0.949	0.956	0.948
50;stemAbstract;InfoGain;CNB	0.875	0.980	0.924	0.957	0.920
R;stemAbstract;InfoGain;FLR	0.862	0.912	0.887	0.902	0.953
20;AllVars;GainRatio;HP	0.934	0.802	0.863	0.825	0.980

Table 9
Mc Nemar'test results for the better configurations, comparing with *10;AllVars; GainRatio;FLR*.

Id	χ Mc Nemar statistic	p-value
50;AllVars;GainRatio;HP	47.457	0.00
50;AllVars;GainRatio;FLR	0.533	0.47
50;stemAbstract;InfoGain;CNB	99.849	0.00
R;stemAbstract;InfoGain;FLR	18.317	0.00
20;AllVars;GainRatio;HP	30.533	0.00

Table 10
Best results for SVM configurations.

Id	P	R	F_1	F_2	ROC
R;stemTitle;InfoGain;SVM	0.867	0.526	0.655	0.570	0.761
10;AllVars;GainRatio;SVM	0.861	0.600	0.707	0.639	0.795
20;AllVars;GainRatio;SVM	0.866	0.850	0.858	0.853	0.909
50;AllVars;GainRatio;SVM	0.922	0.952	0.937	0.946	0.936

Instead of the common bag-of-words approach, a novel representation is proposed based on the use of semantic information from biomedical resources such as UMLS, nomenclature rules for naming drugs or MeSH vocabulary. Our main hypothesis is that semantic information is useful to deal with the problem of data sparseness.

We have performed an extensive experimental analysis using a combination of techniques for feature selection and the most important machine learning algorithms for text classification [64]. We have studied the behavior of features in relation with attribute dimensionality reduction when feature selection algorithms are applied, and with their contribution to the final classification results. The best result has been achieved by a Fuzzy Lattice Reasoning classifier, reaching 0.94, 0.95 and 0.98 of F_1, F_2 and ROC area, respectively. We plan to further improve the accuracy of our classification system taking into account the findings of the present work. Furthermore, since many of the articles are also available in full-text, we will include full article analysis in our future research.

Acknowledgements

This research paper is supported by Projects TIN2007-67407-C03-01, S-0505/TIC-0267 and MICINN project TEXT-ENTERPRISE 2.0 TIN2009-13391-C04-03 (Plan I + D + i), as well as for the Juan de la Cierva program of the MICINN of Spain. The authors are grateful to María Segura Bedmar, manager of the Drug Information Center of the Mostoles University Hospital, Spain, for her valuable assistance in the creation and evaluation of the corpus.

References

[1] Imming P, Sinning C, Meyer A. Drugs, their targets and the nature and number of drug targets. Nat Rev Drug Discov 2006;5(10):821–34.
[2] Hopkins AL, Groom CR. The druggable genome. Nat Rev Drug Discov 2002;1(9):727–30.
[3] Wishart DS, Knox C, Guo AC, Shrivastava S, Hassanali M, Stothard P, et al. DrugBank: a comprehensive resource for in silico drug discovery and exploration. Nucleic Acids Research. 2006. <http://www.drugbank.ca>.
[4] Adler AJ. Mechanisms of T cell tolerance and suppression in cancer mediated by tumor-associated antigens and hormones. Curr Cancer Drug Targets 2007;7(1):3.
[5] Bean P. New drug targets for HIV. Clin Infect Dis 2005;41(S1):96–100.
[6] Di Matteo V, Esposito E. Biochemical and therapeutic effects of antioxidants in the treatment of Alzheimer's disease, Parkinson's disease, and amyotrophic lateral sclerosis. Curr Drug Targets CNS Neurolog Disord 2003;2(2):95.
[7] Zheng C, Han L, Yap C, Ji Z, Cao Z, Chen Y. Therapeutic targets: progress of their exploration and investigation of their characteristics. Pharmacolog Rev 2006;58(2):259–79.
[8] Bolcskei H, Farkas B, Kocsis P, Tarnawa I. Recent advancements in antimigraine drug research: focus on attempts to decrease neuronal hyperexcitability. Recent Patents CNS Drug Discov 2009;4(1):14.
[9] Sauve A. Pharmaceutical strategies for activating sirtuins. Curr Pharm Des 2009;15(1):45.
[10] Deal C. Potential new drug targets for osteoporosis. Nature Publishing Group; 2009.
[11] Chen X, Ji Z, Chen Y. TTD: therapeutic target database. Nucleic Acids Res 2002;30(1):412.
[12] Aguero F, Al-Lazikani B, Aslett M, Berriman M, Buckner FS, Campbell RK, et al. Genomic-scale prioritization of drug targets: the TDR Targets database. Nat Rev Drug Discov 2008;7(11):900–7.
[13] Gunther S, Kuhn M, Dunkel M, Campillos M, Senger C, Petsalaki E, et al. SuperTarget and Matador: resources for exploring drug–target relationships. Nucleic Acids Res 2007.
[14] Rebholz-Schuhmann D, Kirsch H, Arregui M, Gaudan S, Riethoven M, Stoehr P. EBIMed – text crunching to gather facts for proteins from Medline. Bioinformatics 2007;23(2):e237.
[15] Boutet E, Lieberherr D, Tognolli M, Schneider M, Bairoch A. UniProtKB/Swiss-Prot. Meth Mol Biol 2007;406:89–112.
[16] Harris M, Clark J, Ireland A, Lomax J, Ashburner M, Foulger R, et al. The Gene Ontology (GO) database and informatics resource. Nucleic Acids Res 2004;32:D258 (Database issue).
[17] Lindberg DA, Humphreys BL, McCray AT. The Unified Medical Language System. Meth Inform Med 1993;32(4):281. <http://www.nlm.nih.gov/research/umls/>.
[18] Lipscomb CE. Medical subject headings (MeSH). Bull Med Libr Assoc 2000;88(3):265. <http://www.ncbi.nlm.nih.gov/mesh>.
[19] Aronson A. Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program. In: Proceedings of the AMIA Symposium. American Medical Informatics Association; 2001. p. 17.
[20] Wishart DS, Knox C, Guo AC, Cheng D, Shrivastava S, Tzur D, et al. DrugBank: a knowledgebase for drugs, drug actions and drug targets. Nucleic Acids Res 2008;36:D901–6 (Database issue).
[21] Hoffmann R, Valencia A. Implementing the iHOP concept for navigation of biomedical literature. Bioinformatics 2005;21(90002).
[22] Yeh A, Hirschman L, Morgan A. Background and overview for KDD Cup 2002 task 1: information extraction from biomedical articles. ACM SIGKDD Explorations Newslett 2002;4(2):87–9.
[23] Eppig JT, Bult CJ, Kadin JA, Richardson JE, Blake JA. The Mouse Genome Database (MGD): from genes to mice – a community resource for mouse biology. Nucleic Acids Res 2005;33(Database Issue):D471.
[24] Hersh W, Bhuptiraju RT, Ross L, Johnson P, Cohen AM, Kraemer DF. TREC 2004 genomics track overview. In: Proceedings of the 13th text retrieval conference; 2004. <http://ir.ohsu.edu/genomics/>.
[25] Hersh W, Cohen A, Yang J, Bhupatiraju RT, Roberts P, Hearst M. TREC 2005 genomics track overview. In: Proceedings of the 14th text retrieval conference (TREC 2005); 2005.
[26] Dayanik A, Fradkin D, Genkin A, Kantor P, Lewis DD, Madigan D, et al. DIMACS at the TREC 2004 genomics track. In: Proceedings of the 13th text retrieval conference (TREC 2004); 2004.
[27] Nakov P, Schwartz A, Stoica E, Hearst M. BioText Team experiments for the TREC 2004 Genomics track. In: Proceedings of the the thirteenth text retrieval conference, TREC; 2004.
[28] Cohen A, Bhupatiraju R, Hersh W. Feature generation, feature selection, classifiers, and conceptual drift for biomedical document triage. In: Proceedings of the thirteenth text retrieval conference: TREC; 2004.
[29] Ando RK, Dredze M, Zhang T. TREC 2005 genomics track experiments at IBM Watson. In: Proceedings of the fourteenth text retrieval conference proceedings (TREC 2005); 2005.
[30] Krallinger M, Valencia A. Evaluating the detection and ranking of protein interaction relevant articles: the BioCreative Challenge Interaction Article Sub-Task (IAS). In: Proceedings of second Biocreative challenge evaluation workshop. <http://www.biocreative.org/news/chapter/biocreative-ii/>; 2007. p. 29–39.
[31] Krallinger M, Leitner F, Valencia A. The BioCreative II.5 challenge overview. In: Proceedings of the BioCreative II. 5 Workshop 2009 on Digital Annotations; 2009. <http://www.biocreative.org/news/biocreative-ii5/>.
[32] Alex B, Grover C, Haddow B, Kabadjov M, Klein E, Matthews M, et al. Automating curation using a natural language processing pipeline. Genome Biol 2008;9(Suppl. 2):S10.
[33] Kolchinsky A, Abi-Haidar A, Kaur J, Hamed A, Rocha LM. Classification of protein-protein interaction documents using text and citation network

features. In: Proceedings of the BioCreative II. 5 Workshop 2009 on Digital Annotations; 2009. p. 34.
[34] Duda S, Aliferis C, Miller R, Statnikov A, Johnson K. Extracting drug–drug interaction articles from MEDLINE to improve the content of drug databases. In: AMIA annual symposium proceedings, vol. 2005. American Medical Informatics Association; 2005. p. 216.
[35] Zhang X, Zhou X, Hu X. Semantic smoothing for model-based document clustering. In: IEEE international conference on data mining (ICDM'06); 2006.
[36] Drysdale RA, Crosby MA, et al. FlyBase: genes and gene models. Nucleic Acids Res 2005;33(Database Issue):D390. <http://flybase.org/>.
[37] Hermjakob H, Montecchi-Palazzi L, Lewington C, Mudali S, Kerrien S, Orchard S, et al. IntAct: an open source molecular interaction database. Nucleic Acids Res 2004;32(Database Issue):D452. <http://www.ebi.ac.uk/intact/main. xhtml>.
[38] Zanzoni A, Montecchi-Palazzi L, Quondam M, Ausiello G, Helmer-Citterich M, Cesareni G. MINT: a Molecular INTeraction database. FEBS Lett 2002;513(1): 135–40. <http://mint.bio.uniroma2.it/mint/Welcome.do>.
[39] McCray AT, Burgun A, Bodenreider O. Aggregating UMLS semantic types for reducing conceptual complexity. In: Proceedings of Medinfo 2001 world congress on medical informatics, vol. 84. IOS-Press; 2001. p. 216–220.
[40] Segura-Bedmar I, Martínez P, Segura-Bedmar M. Drug name recognition and classification in biomedical texts. A case study outlining approaches underpinning automated systems. Drug Discov Today 2008;13(17–18): 816–23.
[41] Drugs E, Policy M. The use of stems in the selection of International Nonproprietary Names (INN) for pharmaceutical substances; 2007.
[42] Quinlan JR. C4. 5: programs for machine learning. Morgan Kaufmann; 1993.
[43] John GH, Langley P. Estimating continuous distributions in Bayesian classifiers. In: Proceedings of the 11th conference on uncertainty in artificial intelligence. Morgan Kaufmann; 1995. p. 338–45.
[44] Rennie JD, Shih L, Teevan J, Karger D. Tackling the poor assumptions of naive bayes text classifiers. In: Proceedings of the 20th international conference on machine learning; 2003. p. 616-23.
[45] Bouckaert RR, of Waikato U, of Computer Science D. Bayesian network classifiers in weka. Dept. of Computer Science, University of Waikato; 2004.
[46] Su J, Zhang H, Ling CX, Matwin S. Discriminative parameter learning for Bayesian networks. In: Proceedings of the 25th international conference on machine learning. New York (NY, USA): ACM; 2008. p. 1016–23.
[47] Wu X, Kumar V, Ross, Ghosh J, Yang Q, Motoda H, et al. Additive logistic regression: a statistical view of boosting. Ann Stat 2000;28(2):337–74.
[48] Landwehr N, Hall M, Frank E. Logistic model trees. Mach Learn 2005;59(1): 161–205.
[49] Sumner M, Frank E, Hall M. Speeding up logistic model tree induction. Lect Notes Comput Sci 2005;3721:675.
[50] Athanasiadis IN, Kaburlasos VG, Mitkas PA, Petridis V. Applying machine learning techniques on air quality data for real-time decision support. In: First international NAISO symposium on information technologies in environmental engineering (ITEE'2003). Poland: Gdansk; 2003.
[51] Kaburlasos VG, Athanasiadis IN, Mitkas PA. Fuzzy lattice reasoning (FLR) classifier and its application for ambient ozone estimation. Int J Approx Reason 2007;45(1):152–88.
[52] Hsu CW, Chang CC, Lin CJ, et al. A practical guide to support vector classification. a; 2003.
[53] Witten IH, Frank E. Data mining: practical machine learning tools and techniques with Java implementations. Morgan Kaufmann; 2005.
[54] Chang CC, Lin CJ. LIBSVM: a library for support vector machines; 2001.
[56] Fawcett T. An introduction to ROC analysis. Pattern Recogn Lett 2006;27(8): 861–74.
[57] Davis J, Goadrich M. The relationship between precision–recall and ROC curves. In: 23rd international conference on Machine learning; 2006. p. 240–248.
[59] Hall MA. Correlation-based feature selection for Machine Learning. University of Waikato, Department of Computer Science; 1999.
[60] Hall MA, Smith LA. Practical feature subset selection for machine learning. Comput Sci 1998;98:4–6.
[61] Kaburlasos VG, Athanasiadis IN, Mitkas PA. Fuzzy lattice reasoning (FLR) classifier and its application for ambient ozone estimation. Int J Approx Reason 2007:152–88.
[62] Piedra-Fernandez JA, Canton-Garbin M, Guindos-Rojas F. Application of fuzzy lattice neurocomputing (FLN) in ocean satellite images for pattern recognition. Stud Comput Intell 2007;67:215–32.
[63] Dietterich TG. Approximate statistical tests for comparing supervised classification learning algorithms. Neural Comput 1998;10(7):1895–923.
[64] Wu X, Kumar V, Quinlan JR, Ghosh J, Yang Q, Motoda H, et al. Top 10 algorithms in data mining. Knowledge Inform Syst 2008;14(1):1–37.
[65] Kononenko I. Estimating attributes: analysis and extensions of RELIEF. Lect Notes Comput Sci 1994;784:171–82.
[66] Robnik-Sikonja M, Kononenko I. An adaptation of Relief for attribute estimation in regression. In: Proceedings in Machine learning International workshop THEN conference; 1997. p. 296–304.
[67] Abramowitz M, Stegun I. Handbook of mathematical functions with formulas, graphs, and mathematical tables. New York: Dover; 1964.