Towards Unsupervised Text Classification Leveraging Experts and Word Embeddings

Zied Haj-Yahia
Capgemini Invent
zied.haj-yahia@capgemini.com

Adrien Sieg
Capgemini Invent
adrien.sieg@capgemini.com

Léa A. Deleris
BNP Paribas
lea.deleris@bnpparibas.com

Abstract

Text classification aims at mapping documents into a set of predefined categories. Supervised machine learning models have shown great success in this area but they require a large number of labeled documents to reach adequate accuracy. This is particularly true when the number of target categories is in the tens or the hundreds. In this work, we explore an unsupervised approach to classify documents into categories simply described by a label. The proposed method is inspired by the way a human proceeds in this situation: It draws on textual similarity between the most relevant words in each document and a dictionary of keywords for each category reflecting its semantics and lexical field. The novelty of our method hinges on the enrichment of the category labels through a combination of human expertise and language models, both generic and domain specific. Our experiments on 5 standard corpora show that the proposed method increases F1-score over relying solely on human expertise and can also be on par with simple supervised approaches. It thus provides a practical alternative to situations where low-cost text categorization is needed, as we illustrate with our application to operational risk incidents classification.

Ce n'est pas totalement vrai parce que les humains comprennent le sens de la phrase dans son ensemble

1 Introduction

Document classification is a standard task in machine learning (Joachims, 1999; Sebastiani, 2002). Its applications span a variety of "use cases and contexts, e.g., email filtering, news article clustering, clinical document classification, expert-question matching". The standard process for text categorization relies on supervised and semi-supervised approaches.

The motivation for the present effort comes from the banking sector, in particular the management of operational risks. This category of risks corresponds to the broad set of incidents that are neither credit nor market risk and includes issues related to internal and external fraud, cybersecurity, damages on physical assets, natural disasters, etc. The practical management of operational risk is partially based on the management of a dataset of historical operational risk incidents where each incident is described in details and that is shared on a regular basis with regulators.

Historically, all incident reports have been mapped to about twenty categories of risk issued from the regulator. However, from an operational perspective, a higher number of risk categories is relevant to better capture the nuances around the incidents and enable relevant comparisons. This led to the creation of a new internal risk taxonomy of risk composed of 264 categories, each described by a label (a few words). To make it operational, the stock of all internal and external incident reports had to be classified into categories from the new internal taxonomy. However, since it had never been used before, we had no labeled samples readily available. As hundreds of thousands of incidents had to be processed, text classification seemed a promising approach to assist in that mapping task. Indeed, given the specificity of the domain and the lack of availability of experts, it was not conceivable to obtain many labeled examples for each category as would be required for supervised approaches.

This is the issue addressed in this paper where describe our work towards an unsupervised approach to classify documents into a set of categories described by a short sentence (label). While the inspiration of this paper is the classification of incident reports in operational risk, our approach aims to be readily transferable to other domains. For that purpose, we tested it on standard text classification corpora.

The underlying idea is altogether simple. We

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 371–379
Florence, Italy, July 28 - August 2, 2019.

emulate the approach that a domain expert would follow to manually assign an input document (incident report, client review, news article, etc.) to a given category. Specifically this entails developing an understanding of the categories semantic fields and then, for each document, to classify it into the closest category. The novelty of our method hinges on the diversity of enrichment techniques of the categories label, including expert input that assists the semantic expansion and the use of word embeddings, both generic and domain specific.

The remainder of this paper is organized as follows. In Section 2, we provide an overview of the relevant literature. Section 3 contains a detailed description of our approach. Sections 4 and 5 describe the results of its application to standard corpora and operational risks incidents respectively. We conclude in Section 6.

2 Related Work

In this review of relevant work, we focus predominantly on techniques that have been proposed to overcome the requirement of having a large number of annotated data for standard text classification techniques. Overall, the majority of approaches focus on generating labeled examples without full manual annotation.

Those include semi-supervised techniques that seek to leverage a small set of labeled documents to derive labels for the remainder of the corpus. For instance, Nigam et al. (2000) propose to follow the Expectation-Maximization (EM) algorithm by iteratively using the set of labeled data to obtain probabilistically-weighted class labels for each unlabeled document and then training a classifier on the complete corpus based on those annotations. This process is repeated until convergence of the log likelihood of the parameters given observed data. Other approaches attempt to automatically derive labels without any starting set of annotations. For instance, Turney (2002) classifies a review as recommended or not recommended by computing the pointwise mutual information of the words in the review with a positive reference word (excellent) and with a negative reference word (poor) using search engine results as a proxy for a reference corpus. Another example is Ko and Seo (2000) who leverage an initial set of manually provided keywords for each target category to derive labels. Based on those keywords, they look for representative sentences in the corpus to support label assignment. Finally, Yang et al. (2013) make use of wikipedia as background knowledge to assemble representative set of words for each category label via topic modeling and use them to annotate the unlabeled documents. In a similar way, Miller et al. (2016) represent each target category as a TF-IDF (term-frequency/inverse document frequency) vector obtained from Wikipedia and then use this category representation as an informed prior to Latent Dirichlet Allocation (LDA), an unsupervised algorithm that finds the topics that best satisfy the data given the priors. The occurrence of these topics in a document can be used as a noisy label for that document.

Our approach differs in spirit in the sense that our objective is not to construct surrogate labels so that we can apply a machine learning classifier to our unlabeled data. By contrast, we opted for a fully unsupervised method which hinges on computing a similarity metric between documents and target categories. To that end, a richer representation of category labels is derived. The method that were proposed by Yang et al. (2013); Miller et al. (2016) could be adapted to align with our perspective (by removing the classification step). Other examples of unsupervised approach include Rao et al. (2006) which defined the label of documents based on a k-means word clustering. They select a set of representative words from each cluster as a label and derive a set of candidate labels. An input document vector is then assigned to the label vector that maximizes the norm of the dot-product. While this approach performs well when there are no categories specified as input, e.g., social listening, trend monitoring, topic modeling, it is less likely to do so with a set of predefined target categories where it is difficult to steer word clusters to categories of interest and, critically, to ensure the full coverage of target categories (new internal taxonomy of risk in our practical case).

Finally, our method makes use of word embeddings as a mean to enrich category label via semantic expansion. As far as we know, word embeddings have been used to improve text classification performance through their application as a document representation technique. In Liu et al. (2018), the authors show that task oriented embeddings, which penalise outputs where the representative words of a category are close to the

representative words of another category, outperform general domain embeddings. As we do not have any labeled data, this approach is not directly relevant to our problem setting.

3 Method

Our approach for unsupervised text classification is based on the choice to model the task as a text similarity problem between two sets of words: One containing the most relevant words in the document and another containing keywords derived from the label of the target category. While the key advantage of this approach is its simplicity, its success hinges on the good definition of a **dictionary of words** for each category.

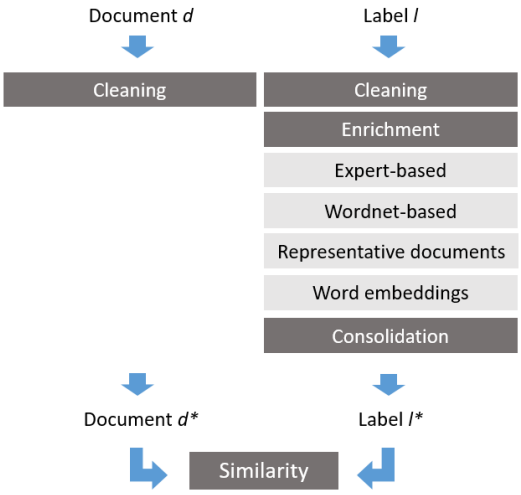

Figure 1: Overview of our Method

Figure 1 provides an overview of the main steps included in our method. On the document side, we simply perform standard *cleaning* steps. On the category labels side, besides the same initial processing, we implement a series of *enrichment* steps so as to iteratively expand label dictionaries. Before proceeding to the comparison of documents and labels via a similarity metric, we have added a *consolidation* step which considers all expanded label dictionaries and makes adjustments so that they are as discriminating as possible. We compare documents and labels by computing a *similarity metric* between cleaned documents and dictionaries. We provide further details into each of these main steps in the following subsections. In terms of notation, we refer to the unlabeled corpus as C, its vocabulary as V and and assume that we have M text categories to which documents in C need to be mapped.

3.1 Cleaning Steps

Cleaning of either documents or category labels is done as follows: After tokenization, we start by replacing a list of common abbreviations, e.g., *Mgt, Mngt, IT, ATM* provided by business with their associated expansions. Similarly we spell out negative contractions. We then remove uninformative tokens including (i) isolated and special characters such as *i, a, o, op, @, **, (ii) punctuation (iii) stopwords (based on stopword lists from *NLTK's list of english stopwords*, *scikit-learn version 0.18.2*, *spaCy version 1.8.2*) (iv) common words across documents such as *risky, dangerous*, based on the highest Term Frequency (top 3 %) (v) uncommon words, i.e., top 3 % in terms of Inverse Term Frequency (vi) specific tokens such as dates, nationalities, countries, regions, bank names. For instance, to extract dates, we use both regular expression and fuzzy matching to identify all sorts of date-like strings (e.g., February can also be written as Feb or Febr). Regarding nationalities and bank names, we combined different lists coming from Wikipedia, business experts and fuzzy matching (e.g., BNP Paribas could be found as BNP, BNPParibas, BNP Securities, BNP Trading, BNP Group, etc.). As the taxonomy is designed to be universal, such tokens are not relevant to the text classification task and are thus removed.

+ version étendue des abbréviations

- contractions négatives

- tokens non informatifs

- mots les plus communs

- mots les plus rares

- constantes non informatives

To give a concrete example, the following snippet of operational incident *"On 18 June 2013 the US Commodity Futures Trading Commission (CFTC) fined ABN AMRO Clearing Chicago USD 1 million (EUR 748,000) for failing to segregate or secure sufficient customer funds, failing to meet the minimum net capital requirements, failing to maintain accurate books and records, and failing to supervise its employees..."* would have been transformed into *"fine fail segreg secur suffici custom fund fail meet minimum net capit requir fail maintain accur book record fail supervis employe.."*

+Stemming

3.2 Enrichment

As mentioned previously, once we have clean labels, we make a series of enrichment steps.

First, we make use of **Expert Knowledge**, i.e., a human expert is asked to provide 3 to 5 additional words for each label. While this constitutes a small amount of manual effort, there are multiple ways to approximate this task without human intervention, for example, by querying Wikipedia or

the web with the category name and performing token counts over retrieved entries. Before proceeding to the next enrichment step, we also add to the label dictionaries all the spelling variants of the expert-provided words that can be found in the document corpus. We also remove any word whose stem is not in the document corpus.

Second, we leverage **WordNet** (Fellbaum, 1998) to obtain knowledge-based synonyms. For every word obtained in the previous step, we add to the label dictionary all the associated synonym sets (English nouns, verbs, and adjectives). Again, once this step is completed, we remove all words where the stem is not in the vocabulary V.

Third, we bootstrap the label dictionary obtained upon this point by making use of **representative documents**. A representative sentence for a given category is defined by Ko and Seo (2000) as a sentence in the document corpus that contains manually pre-defined keywords of the category in its content words. In this work, we extend this definition to apply to documents instead of sentences and to include all categories' keywords obtained at this stage. Therefore we calculate a similarity score between each pair of input document - category label keywords using cosine distance and Latent Semantic Analysis. The text similarity metric will be details in section 3.4. For this step, we use an empirically identified similarity threshold (70%). Then, for each identified representative document, we add all its words to the label dictionary.

seul très élevé de similarité

Finally, we make use of **word embeddings** (Bengio et al., 2003; Mikolov et al., 2013a,b) to further capture semantically similar words to the ones belonging to each label dictionary. We first proceed with pre-trained models which enable to identify semantically similar words used in the general domain. In our case, we used Glove[1] (Pennington et al., 2014), The model is pre-trained on a corpus using Wikipedia2014 and Gigaword5, with a 330 vocabulary of the top 400,000 most frequent words and a context window size of 10.

Furthermore, we also seek to obtain similar words as used in the specific domain of the corpus. Since the neighbors of each keyword are semantically related in embedding space (Mikolov et al., 2013b), we train a Word2Vec model, trained on all input documents cleaned then joined together. In this work, we tested its two main architectures:

pourquoi n'avoir pas utiliser aussi Glove?

[1] https://nlp.stanford.edu/projects/glove/

Continous Bag of words (CBOW) that predicts a word based on its context defined by a sliding window of words and Skip-Gram (SG) which predicts the context given the target word. Experimental settings will be detailed in section 4.3.

3.3 Consolidation

Once all labels have been associated with dictionaries, we perform a final step in order to reduce keyword overlap among all dictionaries. In essence, we favor words that are representative (salient) for the category in the sense that they have the ability to distinguish the category label from the other categories.

We adapt the Function-aware Component (FAC) originally used in supervised document classification (Liu et al., 2018).

$$FAC(w,c) = \frac{TF(w,c) - \frac{1}{M}\sum_{1\leq k\leq M} TF(w,k)}{var(TF_{-c}(w))} \quad (1)$$

where $TF_{-c}(w)$ is the collection of term frequencies except the c-th category and $var()$ is the variance.

The consolidation step consists in computing the above metric for every word in the label dictionaries and to filter out those whose associated metric is below a given threshold. This latter threshold depends on two main constraints: The maximum number of categories that contain a given word and the minimum word frequency in the label dictionaries. Regarding the first constraint, in our practical case of operational risk taxonomy, we have 264 target categories that could be grouped into 16 broad categories: cyber-security, fraud, compliance, human resources, etc. Thresholds are determined so as to tolerate overlap within each broad category and to minimize it outside. More generally, we start by identifying the maximum number of semantically similar categories, i.e., where we would expect some overlap and we set the threshold consequently. By construction, keywords in a given dictionary occur at least one time. We decided not to set an additional constraint on word frequency per category label so as to keep highly specific words with a low frequency, generally captured by the Word2vec model trained on the input corpus.

3.4 Text Similarity Metric

Once documents and labels have been processed as described previously, we assign a label to a document by identifying the label to which it is most similar. Our evaluation of similarity is based on Latent Semantic Analysis (to avoid the pitfalls of literal term matching) and cosine similarity on the output LSA vectors. Before applying LSA, we start by stemming all the words using Porter stemmer.

We feel that similarities between documents and labels can be more reliably estimated in the reduced latent space representation than in the original representation. The rationale is that documents which share frequently co-occurring terms will have a similar representation in the latent space, even if they have no terms in common. LSA thus performs some sort of noise reduction and has the potential benefit to detect synonyms as well as words that refer to the same topic.

4 Experiments

4.1 Datasets

In order to evaluate our approach, we conduct experiments on five standard text classification corpora, described listed in Table 1. As we use an unsupervised approach for text classification, we make use of the whole corpus of each dataset by aggregating training and test sets.

Datasets	#Documents	#Classes
20NewsGroup	18,846	20
AG's Corpus	126,764	4
Yahoo-Answers	1,460,000	10
5AbstractsGroup	7,497	5
Google-Snippets	10,059	8

Table 1: Statistics of the five mainstream datasets for text classification.

We describe each corpus briefly: (1) The **20NewsGroup**[2] dataset consists of 18,846 news articles divided almost evenly among 20 different UseNet discussion groups. Some of the newsgroups are closely related (e.g., comp.sys.ibm.pc.hardware and comp.sys.mac.hardware). While each document may discuss multiple topics, it needs to be assigned to a single category. (2) The **AG's Corpus of news articles**[3] is a collection of more than 1 million news articles. We used the version created by Zhang et al. (2015) who selected 4 largest classes from AG news corpus on the web with each instance containing class index, title and description fields. (3) The **Yahoo-Answers**[4] corpus contains 4,483,032 questions and their corresponding answers from Yahoo! Answers service as of 10/25/2007. We used the version constructed by Zhang et al. (2015) using 10 largest main categories and the best answer content from all the answers. (4) The **5AbstractsGroup**[5] dataset is a collection of academic papers from five different domains collected from Web of Science namely, business, artificial intelligence, sociology, transport and law. We extracted the abstract and title fields of each paper as a document. (5) The **Google-Snippets**[6] dataset contains the web search results related to 8 different domains such as business, computers and engineering.

4.2 Configurations and Baseline Methods

We apply multiple variants of our method to each of the above corpora. Note first that using representative documents (Section 3.2) to enrich label dictionaries is suitable for categories whose labels take the form of a structured sentence containing more than 10 words before cleaning. In the application to operational risk incidents (Section 5), it allowed to enrich 13% of dictionaries. In the standard text classification datasets used in our experiments, category labels contain less than 5 words so representative documents were not relevant in the enrichment process. Thus none of the configurations discussed in this section include this step.

Overall, in addition to the full pipeline, which we refer to as all keywords, we also investigated whether semantic expansion solely through word embeddings could improve performance. We thus tested with either generic embeddings (pre-trained Glove) or corpus-based embeddings (Word2Vec). Finally, for each configuration, we tested with and without the function aware component (FAC) for consolidation of the label dictionaries.

We also implemented simple baselines for comparison. On the unsupervised side, (1) we calculated a text similarity score between each docu-

[2] http://qwone.com/ jason/20Newsgroups/

[3] www.di.unipi.it/ gulli/AG_corpus_of_news_articles.html

[4] https://github.com/LC-John/Yahoo-Answers- Topic-Classification-Dataset /tree/master/dataset

[5] https://github.com/qianliu0708/5AbstractsGroup

[6] http://jwebpro.sourceforge.net/data-web-snippets.tar.gz

World	Sports	Business	Science/Technology
Election	Olympic	Company	Laboratory
State	Football	Market	Computers
President	Sport	Oil	Science
Police	League	Consumers	Technology
Politics	Baseball	Exchange	Web
Security	Rugby	Business	Google
War	Tickets	Product	Microsoft
Nuclear	Basketball	Price	Economy
Democracy	Games	Billion	Software
Militant	Championship	Stocks	Investment

Table 2: Example of ten salient words for each category in the AGs Corpus dataset.

ment and the set of expert provided keywords (2) we enriched this list of initial keywords with their synonyms from WordNet. On the supervised side, we use Multinomial Naïve Bayes as a basic baseline where we represented each document as TF-IDF vector (bag of words), cleaned the input corpus in the same way as in our proposed approach and split each dataset into a training set (2/3) and a test set (1/3).

4.3 Experimental Settings

In our method, an offline process is used to extract initial keywords from category labels. For the purpose of testing our approach, we had to emulate human experts ourselves. For each category, one team member added a few keywords based only on label description. Then, we randomly selected 2 or 3 documents for each label that were read by two team members who used them to identify 5 to 10 salient words to be added to each dictionary. In average, we manually added 9 words per label for 20NewsGroup, 17 words for AGs Corpus and Google-Snippets, 11 words for Yahoo-Answers and 14 words for 5AbstractsGroup. We present in Table 2, the output of that process for the AGs Corpus dataset.

Once we identify initial keywords, we make the series of enrichment steps described in section 3.2. For every word in the set of initial keywords, we add all its synonym sets from WordNet as well as the 10 most similar words from Glove, CBOW and Skip-Gram. The average length of label dictionaries obtained from the full enrichment pipeline (which we refer to as all keywords) is 428 words.

We use the word2vec python implementation provided by gensim (Rehurek and Sojka, 2010). For Skip-gram and CBOW, a 10-word window size is used to provide the same amount of raw information. Also words appearing 3 times or fewer are filtered out, 10 workers were used and training was performed in 100 epochs. We chose 300 for the size of all word embeddings, it has been reported to perform well in classification tasks (Mikolov et al., 2013a).

Filtering word dictionaries with the Function-aware Component (FAC) allowed to keep in average 37% of all keywords per label. As described previously, once different versions of label dictionaries have been obtained, we calculate their similarity with input documents using LSA and Cosine distance. The optimal dimension (k) of the latent space depends on the dataset. Optimal k values are typically in the range of 100-300 dimensions (Harman, 1993; Letsche and Berry, 1997). In this work, for each dataset, we set a range of 100-300 values, and we determine the optimal k by maximizing the topic coherence score (Röder et al., 2015).

The multi-class classification performance was evaluated in terms of precision (Prec.), recall (Rec.) and F1-score (F1). All measures are computed based on a weighted average of each class using the number of true instances to determine the weights.

4.4 Results and Discussion

Table 3 summarizes the performance of each of the methods tested on the five corpora that we considered. Overall, the various configurations of our method, all leveraging embeddings for semantic expansion, outperform the simple unsupervised baselines, leading to a doubling of the F1-score for all corpora, the least affected being the 5Abstracts-Group where F1 goes from 38.1 to 68.3 percent, comparing with the all keywords variant of our method.

When focusing on our various configurations, first without the FAC consolidation, we observe that domain specific embeddings alone lead to better performance than generic embeddings alone and this across all corpora and all metrics, except for the Yahoo-Answers dataset. The difference in performance however is not very large, with the exception of 20NewsGroup where F1-score increases from 52.6 with generic embeddings to 61 with domain specific ones. We notice also that combining all enrichments (All keywords) provides a modest increase in performance over embeddings only as shown by the results for Yahoo-Answers, 5AbstractsGroup and Google-Snippets. Finally the use of the consolidation step further

improves performance except for 20NewsGroup where precision increases from 64.7 to 71.1 but recall decreases from 57.8 to 35.6.

Comparing now our best unsupervised performance with the supervised baseline, we observe that the ratio of the best F1-score performance over the supervised baseline performance varies from 0.71 to 1.11 with two datasets yielding ratios above 1. Such results demonstrate the validity of the unsupervised approach as a practical alternative to investing to a cognitively and timely costly annotation effort.

5 Application to Operational Risk Incident Classification

As we described previously, the proposed method stemmed from a specific need in the banking industry where a large number of incidents had to be mapped to a newly defined taxonomy of operational risks. Specifically, it was designed to avoid the tedious and time consuming effort of asking experts to manually review thousands of incidents. An automated - or more precisely assisted - approach also presented the additional benefit of ensuring a higher degree of consistency in the mapping than would have been achieved a team of annotators. In this section, we provide some additional context into this specific task, report the observed performance of our method and discuss some of the specificities of the context.

5.1 Operational Risk Incidents Corpus & Taxonomy

In our application, we were asked to map both internal incidents and external incidents to the new taxonomy. In this paper, we focus on the external incidents for confidentiality reasons. More precisely, our task was to assign a unique category to each one of the 25,000 incidents that was obtained from ORX news. The Operational Risk Exchange (ORX) is a consortium of financial institutions focused on operational risk information sharing. The ORX news service provides publicly reported operational risk loss data to its institutional members.

An incident record is mostly composed of an incident description along with associated meta-information such as geographical indicators, time information and institution affected. We only make use of the incident descriptions. Their average length is 2150 words, with a standard deviation of 1181 words and ranging from 10 words to more than 14000 words.

The target taxonomy is composed of three levels. The first one contains 16 labels and indicates at a very high level the domain of the incidents such as IT, legal, regulatory. The second and third levels contain respectively 69 and 264 levels to add increasing granularity to the incident classification. Figure 2 presents an extract of the taxonomy focused on ICT risk, which is public as it draws upon Article 107(3) of Directive 2013/36/EU2 which aim to ensure the convergence of supervisory practices in the assessment of the information and communication technology (ICT) risk.

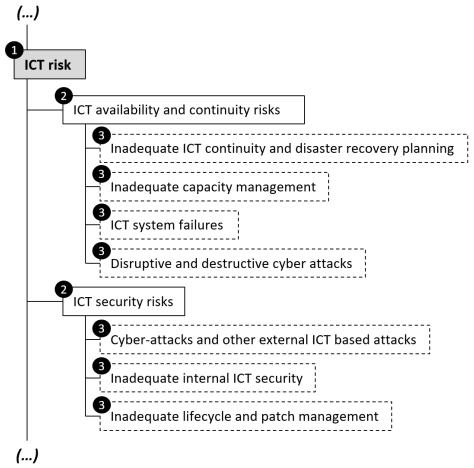

Figure 2: Example of Taxonomy regarding three levels for an ICT incident

Before discussing the results, we thought it would be meaningful to point out some of the characteristics of this application. One natural challenge in real world cases is the lack of unequivocal ground truth. Experts can often identify categories that do not correspond to the input but in the end, they cannot ascertain whether one category should prevail over another unless there is some clear guidelines or convention at the level of the organization. That difficulty is further compounded in our case as most documents are very dense in term of information and become ambiguous. For instance, "*In Japan, a building destruction resulting from a massive earthquake has caused power outage making AMD-based servers unbootable*", could be classified as *Natural Disaster*, *Dysfunctional ICT data processing or handling* or *Destruction / loss of physical assets* among others.

Methods	20NewsGroup			AG's Corpus			Yahoo-Answers			5AbstractsGroup			Google-Snippets		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
Expert keywords	38.0	21.4	27.0	32.9	29.8	31.4	20.4	17.9	19.1	41.7	34.5	37.1	39.4	29.3	33.6
Expert keywords + WordNet	39.2	24.2	27.0	33.7	31.4	32.5	21.8	19.2	20.4	42.5	35.7	38.1	41.6	32.5	36.5
Generic embeddings (Glove)	57.8	48.2	52.6	72.2	72.3	72.2	54.6	50.5	52.5	67.5	66.0	66.7	69.2	66.3	67.7
Corpus based embeddings	64.7	**57.8**	**61.0**	75.1	75.2	75.1	50.4	47.9	49.1	69.0	66.2	67.6	72.4	70.0	71.2
All keywords	62.2	54.2	57.9	75.0	74.9	74.9	54.9	51.9	53.3	69.4	66.9	68.3	71.9	70.1	71.0
FAC-Generic embeddings (Glove)	65.8	34.2	39.6	72.6	72.8	72.5	54.0	52.2	52.1	67.9	61.5	63.2	70.3	65.9	67.5
FAC-Corpus based embeddings	**71.1**	35.6	42.8	**76.8**	**76.6**	**76.6**	59.2	52.7	52.5	70.2	66.8	68.2	72.5	71.3	71.1
FAC-All keywords	66.2	37.8	41.3	74.0	73.8	73.9	**59.3**	**53.9**	**55.7**	**71.5**	**67.3**	**69.7**	**72.9**	**72.8**	**72.8**
Supervised Naïve Bayes	87.1	85.4	85.0	89.8	89.9	89.8	57.2	53.0	49.9	77.5	68.8	65.5	81.8	77.4	77.0

Table 3: Performance of our methods and baseline methods on five standard text classification corpora. Bold numbers indicate the best configurations among the unsupervised approaches. Configurations of our approach do not contain the representative document enrichment step.

Taxonomy level	Prec.	Recall	F1-Score
Level 1	91.80	89.37	90.45
Level 2	86.08	74.80	78.10
Level 3	34.98	19.88	22.95

Table 4: Performance of our Method on the Operational Risk Text Classification Task

5.2 Result

For the purpose of experiment, operational teams (not experts) were asked to provide manual tags for a sample of 989 operational incidents. Table 4 provide the classification results of our approach when compared to those manual annotations, considering all three levels of the taxonomy.

In a second step in the evaluation, an expert was given the difficult task to challenge each time they disagreed the computer and human annotation and determine which was ultimately correct. This exercise indicated that in 32 cases out of 989 operational incidents under consideration for the Level 1 classification, the machine generated category were more relevant (hence correct) than those identified by the operational team.

5.3 Discussion

Given the number of categories, we were satisfied with the level of performance that we observed, especially for Level 1 and Level 2 of the taxonomy. More importantly, as we progress with the follow up exercise of mapping internal incident descriptions, we have evolved from a point where users always mistrust the outcome of the automated classification to a point where users see the suggested mapping from our algorithm as a relevant recommendation.

Our perspective on the success of this method in this particular context is that operational risk is a textbook case where domain specific labels and vocabulary prevail. For instance, technical words such as *forge*, *fictitious*, *bogus*, *ersatz*, or *counterfeit* indicate almost surely that a *Fraudulent Account Opening* operation happened. Most of operational incidents must contain a combination of technical keywords due to their highly operational nature. What the method brings is the ability to combine human expertise through seed words with the strength of the machine which can process and memorize large corpus and derive distributional semantics from it. In this way, the cognitive burden of being exhaustive is lifted from the experts shoulders.

6 Conclusion

In this paper, we present a method for unsupervised text classification based on computing the similarity between the documents to be classified and a rich description of the categories label. The category label enrichment starts with human-expert provided keywords but is then expanded through the use of word embeddings. We also investigated whether a consolidation step that removes non discriminant words from the label dictionaries could have an effect on performance.

We have not explored whether recent advances in word embeddings from instance ELMO (Peters et al., 2018) and BERT (Devlin et al., 2018) could add further benefits. This is certainly an avenue that we seek to explore. However, for our application domain, we expect that it may not lead to increased performance as words are used to a large extent with the same sense across the corpus.

References

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

C Fellbaum. 1998. Wordnet: An on-line lexical database.

Donna K Harman. 1993. *The first text retrieval conference (TREC-1)*, volume 500. US Department of Commerce, National Institute of Standards and Technology.

Thorsten Joachims. 1999. Transductive inference for text classification using support vector machines. In *Icml*, volume 99, pages 200–209.

Youngjoong Ko and Jungyun Seo. 2000. Automatic text categorization by unsupervised learning. In *Proceedings of the 18th conference on Computational linguistics-Volume 1*, pages 453–459. Association for Computational Linguistics.

Todd A Letsche and Michael W Berry. 1997. Large-scale information retrieval with latent semantic indexing. *Information sciences*, 100(1-4):105–137.

Qian Liu, Heyan Huang, Yang Gao, Xiaochi Wei, Yuxin Tian, and Luyang Liu. 2018. Task-oriented word embedding for text classification. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2023–2032.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Timothy Miller, Dmitriy Dligach, and Guergana Savova. 2016. Unsupervised document classification with informed topic models. In *Proceedings of the 15th Workshop on Biomedical Natural Language Processing*, pages 83–91.

Kamal Nigam, Andrew Kachites McCallum, Sebastian Thrun, and Tom Mitchell. 2000. Text classification from labeled and unlabeled documents using em. *Machine learning*, 39(2-3):103–134.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Delip Rao, P Deepak, and Deepak Khemani. 2006. Corpus based unsupervised labeling of documents. In *FLAIRS Conference*, pages 321–326.

Radim Rehurek and Petr Sojka. 2010. Software framework for topic modelling with large corpora. In *In Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. Citeseer.

Michael Röder, Andreas Both, and Alexander Hinneburg. 2015. Exploring the space of topic coherence measures. In *Proceedings of the eighth ACM international conference on Web search and data mining*, pages 399–408. ACM.

Fabrizio Sebastiani. 2002. Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, 34(1):1–47.

Peter D Turney. 2002. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 417–424. Association for Computational Linguistics.

Lili Yang, Chunping Li, Qiang Ding, and Li Li. 2013. Combining lexical and semantic features for short text classification. *Procedia Computer Science*, 22:78–86.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in neural information processing systems*, pages 649–657.