# Analysis of Word Embeddings and Sequence Features for Clinical Information Extraction

**Lance De Vine, Mahnoosh Kholghi, Guido Zuccon, Laurianne Sitbon**
Queensland University of Technology
{l.devine, m1.kholghi, g.zuccon, laurianne.sitbon}@qut.edu.au
**Anthony Nguyen**
The Australian e-Health Research Centre, CSIRO
anthony.nguyen@csiro.au

## Abstract

This study investigates the use of unsupervised features derived from word embedding approaches and novel sequence representation approaches for improving clinical information extraction systems. Our results corroborate previous findings that indicate that the use of word embeddings significantly improve the effectiveness of concept extraction models; however, we further determine the influence that the corpora used to generate such features have. We also demonstrate the promise of sequence-based unsupervised features for further improving concept extraction.

## 1 Introduction

Clinical concept extraction involves the identification of sequences of terms which express meaningful concepts in a clinical setting. The identification of such concepts is important for enabling secondary usage of reports of patient treatments and interventions, e.g., in the context of cancer monitoring and reporting (Koopman et al., 2015), and for further processing in downstream eHealth workflows (Demner-Fushman et al., 2009).

A significant challenge is the identification of concepts that are referred to in ways not captured within current lexical resources such as relevant domain terminologies like SNOMED CT. Furthermore, clinical language is sensitive to ambiguity, polysemy, synonymy (including acronyms) and word order variations. Finally, the information presented in clinical narratives is often unstructured, ungrammatical, and fragmented.

State of the art approaches in concept extraction from free-text clinical narratives extensively apply supervised machine learning approaches. The effectiveness of such approaches generally depends on three main factors: (1) the availability of a considerable amount of high quality annotated data, (2) the selected learning algorithm, and (3) the quality of features generated from the data.

In recent years, clinical information extraction and retrieval challenges like i2b2 (Uzuner et al., 2011) and ShARe/CLEF (Suominen et al., 2013) have provided annotated data which can be used to apply and evaluate different machine learning approaches (e.g., supervised and semi-supervised). Conditional Random Fields (CRFs) (Lafferty et al., 2001) has shown to be the state-of-the-art supervised machine learning approach for this clinical task. A wide range of features has been leveraged to improve the effectiveness of concept extraction systems, including hand-crafted grammatical, syntactic, lexical, morphological and orthographical features (de Bruijn et al., 2011; Tang et al., 2013), as well as advanced semantic features from external resources and domain knowledge (Kholghi et al., 2015).

While there has been some recent work in the application of unsupervised machine learning methods to clinical concept extraction (Jonnalagadda et al., 2012; Tang et al., 2013), the predominant class of features that are used are still handcrafted features.

This paper discusses the application to clinical concept extraction of a specific unsupervised machine learning method, called the Skip-gram Neural Language Model, combined with a lexical string encoding approach and sequence features. Skip-gram word embeddings, where words are represented as vectors in a high dimensional vector space, have been used in prior work to create feature representations for classification and information extraction tasks, e.g., see Nikfarjam et al. (2015) and Qu et al. (2015). The following research questions will be addressed in this paper:

**RQ1:** are word embeddings and sequence level representation features useful when using CRFs for clinical concept extraction?

**RQ2:** to what extent do the corpora used to gener-

ate such unsupervised features influence the effectiveness?

Question one has been partially addressed by prior work that has shown word embeddings improve the effectiveness of information extraction systems (Tang et al., 2015; Nikfarjam et al., 2015). However, we further explore this by considering the effectiveness of sequence level features, which, to the best of our knowledge, have not been investigated in clinical information extraction.

## 2 Related Work

The two primary areas that relate to this work include *(a)* methods for clinical concept extraction, and *(b)* general corpus based approaches for learning word representations.

### 2.1 Clinical Information Extraction

The strong need for effective clinical information extraction methods has encouraged the development of shared datasets such as the i2b2 challenges (Uzuner et al., 2011) and the ShARe/CLEF eHealth Evaluation Lab (Suominen et al., 2013); which in turn have sparked the development of novel, more effective clinical information extraction methods. For example, de Bruijn et al. (2011) used token, context, sentence, section, document, and concept mapping features, along with the extraction of clustering-based word representation features using Brown clustering; they obtained the highest effectiveness in the i2b2/VA 2010 NLP challenge. In the same challenge, Jonnalagadda et al. (2012) leveraged distributional semantic features along with traditional features (dictionary/pattern matching, POS tags). They used random indexing to construct a vector-based similarity model and observed significant improvements.

Tang et al. (2013) built a concept extraction system for ShARe/CLEF 2013 Task 1 that recognizes disorder mentions in clinical free text, achieving the highest effectiveness amongst systems in the challenge. They used word representations from Brown clustering and random indexing, in addition to a set of common features including token, POS tags, type of notes, section information, and the semantic categories of words based on UMLS, MetaMap, and cTAKEs.

Tang et al. (2014) extracted two different types of word representation features: (1) clustering-based representations using Brown clustering, and (2) distributional word representations using random indexing. Their findings suggest that these word representation features increase the effectiveness of clinical information extraction systems when combined with basic features, and that the two investigated distributional word representation features are complementary.

Tang et al. (2014), Khabsa and Giles (2015) and Tang et al. (2015) investigated the effect of three different types of word representation features, including clustering-based, distributional and word embeddings, on biomedical name entity recognition tasks. All developed systems demonstrated the significant role of word representations in achieving high effectiveness.

### 2.2 Corpus Based Methods for Word Representations

Brown clustering (Brown et al., 1992) has probably been the most widely used unsupervised method for feature generation for concept extraction. Both random indexing (Kanerva et al., 2000) and word embeddings from neural language models, e.g., Mikolov et al. (2013), have also been used recently, in part stimulated by renewed interest in representation learning and deep learning. Some of the more notable contributions to the use of word representations in NLP include the work of Turian et al. (2010) and Collobert et al. (2011). Since their inception, Skip-gram word embeddings (Mikolov et al., 2013) have been used in a wide range of settings, including for unsupervised feature generation (Tang et al., 2015). There have also been recent applications of convolutional neural nets to lexical representation. For example, Zhang and LeCun (2015) demonstrated that deep learning can be applied to text understanding from character-level inputs all the way up to abstract text concepts, using convolutional networks.

## 3 Features

We start by examining a set of baseline features that have been derived from previous work in this area. We then turn our attention to unsupervised features to be used in this task and we propose to examine features based on word embeddings, lexical vectors and sequence level vectors. These features will then be tested to inform a CRFs learning algorithm, see Figure 1.

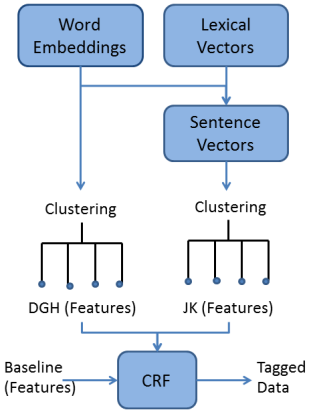


Figure 1: Feature generation process and their use in concept extraction.

### 3.1 Baseline Features

We construct a baseline system using the following baseline feature groups, as described by Kholghi et al. (2015):

- A: Orthographical (regular expression patterns), lexical and morphological (suffixes/prefixes and character n-grams), contextual (window of k words),
- B: Linguistic (POS tags (Toutanova et al., 2003))
- C: External resource features (UMLS and SNOMED CT semantic groups as described by Kholghi et al. (2015)).

### 3.2 Unsupervised Features

The approach we use for generating unsupervised features consists of the following two steps:

1. Construct real valued vectors according to a variety of different methods, each described in Sections 3.2.1– 3.2.3.
2. Transform the vectors into discrete classes via clustering, as described in Section 3.2.4.

While real valued feature vectors can be used directly with some CRFs software implementations, they are not supported by all. We have found that transforming our vectors into discrete classes via clustering is reasonably easy. In addition our preliminary experiments did not show advantages to working with real valued vectors.

We use two types of vectors: semantic and lexical. We use the term "semantic" as an overarching term to refer to neural word embeddings as well as other distributional semantic representations such as those derived from random indexing. The semantic vectors encode a combination of semantic and syntactic information, as distinct to lexical vectors which encode information about the distribution of character patterns within tokens. We find that lexical vectors identify lexical classes within a corpus and are particular useful for corpora where there are many diverse syntactic conventions such as is the case with clinical text.

#### 3.2.1 Semantic Vectors

To construct semantic vectors we use the recently proposed Skip-gram word embeddings. The Skip-gram model (Mikolov et al., 2013) constructs term representations by optimising their ability to predict the representations of surrounding terms.

Given a sequence $\mathcal{W} = \{w_1, \ldots, w_t, \ldots, w_n\}$ of training words, the objective of the Skip-gram model is to maximise the average log probability

$$\frac{1}{2r} \sum_{i=1}^{2r} \sum_{-r \leq j \leq r, j \neq 0} \log p(w_{t+j} | w_t) \qquad (1)$$

where $r$ is the context window radius. The context window determines which words are considered for the computation of the probability, which is computed according to

$$p(w_O | w_I) = \frac{\exp(v_{w_O}^\top v_{w_I})}{\sum_{w=1}^{W} \exp(v_w{}^\top v_{w_I})} \qquad (2)$$

where the $v_{w_I}$ and $v_{w_O}$ are vector representations of the input and output (predicted) words. The value (2) is a normalized probability because of the normalization factor $\sum_{w=1}^{W} \exp(v_w^\top v_{w_I})$. In practice, a hierarchical approximation to this probability is used to reduce computational complexity (Morin and Bengio, 2005; Mikolov et al., 2013).

At initialisation, the vector representations of the words are assigned random values; these vector representations are then optimised using gradient descent with decaying learning rate by iterating over sentences observed in the training corpus.

#### 3.2.2 Lexical Vectors

Various approaches have been previously used to encode lexical information in a distributed vector representation. A common idea in these approaches is the hashing and accumulation of n-grams into a single vector. This is sometimes referred to as string encoding and is used in a variety of applications, including text analysis and bio-informatics (Buhler, 2001; Buckingham et al., 2014). The approach used here is most similar to the holographic word encoding approach of

Hannagan et al. (2011) and Widdows and Cohen (2014).

To create lexical vectors, we first generate and associate a random vector for each distinct character n-gram that is found in the text. Then, for each token we accumulate the vectors for each n-gram contained within the token. We use uni-grams, bi-grams, tri-grams and tetra-grams, but we also include skip-grams such as the character sequence "a_b" where the underscore is a wild-card placeholder symbol. The n-gram vectors are added together and the resulting vector is normalized.

Lexical feature representation is especially useful when there doesn't exist an easily available semantic representation. Some corpora, such as clinical texts, use an abundance of syntactic conventions, such as abbreviations, acronyms, times, dates and identifiers. These tokens may be represented using a lexical vector such that orthographically similar tokens will have similar vectors. An advantage of the use of these lexical vectors is that they are constructed in a completely unsupervised fashion which is corpus independent and does not rely on the use of hand-crafted rules. This is useful in the application to unseen data where there may exist tokens or patterns that have not been seen within the training set (which would in turn render most hand-crafted rules ineffective).

#### 3.2.3 Sequence Level Vectors

Many models of phrase and sentence representation have recently been proposed for tasks such as paraphrase identification, sentiment classification and question answering (Le and Mikolov, 2014; Kalchbrenner et al., 2014), just to name a few. The simple approach adopted in this paper makes use of both semantic and lexical vectors.

To form sequence level vectors, we accumulate the word embeddings for each token in a phrase or sentence. A token is ignored if it does not have an associated word embedding. The lexical vectors for each token in a sequence are also accumulated. Both types of vectors, semantic and lexical, are normalized. We then concatenate the vectors and normalize again.

From time to time, some of the tokens within short text sequences may not be associated to word embeddings. In such a case the sequence is represented entirely with its accumulated lexical vectors. In this paper we evaluate the effectiveness of sentence and bi-gram phrase vectors.

#### 3.2.4 Clustering Methodology

In our approach, the real valued vector representations obtained employing the methods above are then transformed into discrete classes. To cluster these vectors, we use K-means++ (Arthur and Vassilvitskii, 2007) with Euclidean distance using a range of different granularities akin to how multiple levels of representations are generally used in Brown clustering.

Clustering of vectors is performed on a training dataset. When a model is applied to unseen data, the representation for an unseen item is projected into the nearest cluster obtained from the training data, and a feature value is assigned to the item. We experimented with different strategies for assigning feature identifiers to clusters including (a) a simple enumeration of clusters, and (b) a reduced feature space in which only clusters containing a majority of members with the same configuration of concept labels (from training data) are given an incrementing feature number. Method (b) did not improve results and so we only report the outcomes of method (a). Clustering iterations were terminated at 120 iterations. Table 1 and 2 show examples of word and sentence clusters obtained from a clinical corpus.

## 4 Experimental Setup

To evaluate the feature groups studied in this paper, we use the annotated train and test sets of the i2b2/VA 2010 NLP challenge (Uzuner et al., 2011). We evaluate the effectiveness of concept extraction systems using Precision, Recall and F1-measure. Evaluation measures are computed on the i2b2 test data using MALLET's multi-segmentation evaluator (McCallum, 2002) as per the experimental setup of (Kholghi et al., 2014).

We compute statistical significance (p-value) using a 5*2 cross validated t-test (Dietterich, 1998) in which we combine both train and test

Table 1: Example of word embedding clusters.

| | |
|---|---|
| $C_1$ | prediabetes, insulin-dependant, endocrine., early-onset, type-2 |
| $C_2$ | flank/right, extremity/lower, mid-to-lower, extremity/right |
| $C_3$ | knife, scissors, scalpel, clamp, tourniquet |
| $C_4$ | instructed, attempted, allowed, refuses, urged |
| $C_5$ | psychosomatic, attention-deficit, delirium/dementia, depression/bipolar |

Table 2: Example of sentence clusters.

| | |
|---|---|
| $C_1$ | Abs Eos , auto 0.1 X10E+09/L<br>ABS Lymphs 2.4 X10E+09 / L<br>ABS Monocytes 1.3 X10E+09 / L<br>Abs Eos , auto 0.2 X10E+09 / L |
| $C_2$ | 5. Dilaudid 4 mg Tablet Sig : ...<br>7. Clonidine 0.2 mg Tablet Sig : ...<br>9. Nifedipine 30 mg Tablet Sustained ...<br>10. Pantoprazole 40 mg Tablet ... |
| $C_3$ | Right proximal humeral fracture status ...<br>Bilateral renal artery stenosis status ...<br>status post bilateral knee replacement ... |

sets, sample 5 subsets of 30,000 sentences, split each subset into train and test, and perform a paired t-test for these 10 subsets.

As supervised machine learning algorithm for concept extraction, we used a linear-chain CRFs model based on the MALLET CRFs implementation and tuned following Kholghi et al. (2014). We use our own implementation of K-means++ for clustering. For creating the Skip-gram word embeddings we use the popular `word2vec` tool (Mikolov et al., 2013), with hierarchical softmax and 5 epochs on the C1 and C2 datasets and 1 epochs on the PM and WK datasets (see below) due to computational constrains.

### 4.1 Corpora

We use four different corpora to generate word embeddings[1]: two clinical (C1 and C2) and two non-clinical (PM and WK); corpora details are reported below and in Table 3:

- C1: (Clinical) composed by the concatenation of the i2b2 train set (Uzuner et al., 2011), MedTrack (Voorhees and Tong, 2011), and the CLEF 2013 train and test sets (Suominen et al., 2013)
- C2: (Clinical) the i2b2 train set (Uzuner et al., 2011)
- PM: (Biomedical) PubMed, as in the 2012 dump[2]
- WK: (Generalist) Wikipedia, as in the 2009 dump (De Vries et al., 2011)

### 4.2 Feature Groups

In addition to the feature groups `A`, `B` and `C` mentioned in Section 3.1, we consider the following feature groups:

[1]Pre-processing involving lower-casing and substitution of matching regular expressions was performed.

[2]`http://mbr.nlm.nih.gov/Download/`

Table 3: Training corpora for word embeddings.

| Corpus | Vocab | Num. Tokens |
|---|---|---|
| C1 | 104,743 | $\approx$ 29.5 M |
| C2 | 11,727 | $\approx$ 221.1 K |
| PM | 163,744 | $\approx$ 1.8 B |
| WK | 122,750 | $\approx$ 415.7 M |

- `D`: Skip-gram clustering features with window size 2 and 5 and 128, 256, 512, 1024 clusters
- `G`: Window of 3 previous and next Skip-gram clustering feature (window size 2) with 1024 clusters
- `H`: Window of 3 previous and next Skip-gram clustering feature (window size 5) with 1024 clusters
- `J`: Sentence features with 1024 clusters
- `K`: Sentence features with 256 clusters
- `L`: Bi-gram phrase features with 512 clusters
- `M`: Bi-gram phrase features with 1024 clusters

## 5 Results and Discussion

In this section, we first study the impact of different feature sets on the effectiveness of the learnt models. We then discuss how different training corpora affect the quality of word embeddings and sequence representations.

### 5.1 Analysis of Baseline Features

Table 4 reports the effectiveness of CRF models built using only the word tokens appearing in the documents (`Word`), and this feature along with different combinations of baseline features (`A`, `B`, `C`). These results show that feature group `A` (orthographical, lexical, morphological, and contextual features) provides significantly higher effectiveness compared to other individual feature groups. Semantic features (group `C`) also achieve reasonably high effectiveness compared to the use of `Word` features alone. However, POS tags (group `B`) provide inferior effectiveness. Indeed, when feature group `B` is used in combination with either `A` or `C`, no significant differences are observed compared to using `A` or `C` alone: POS tags do not improve effectiveness when combined with another, single feature group. It is the combination of all baseline features (`ABC`), instead, that provides the highest effectiveness.

Table 4: Results for baseline features. Statistically significant improvements ($p<0.05$) for F1 when compared with Word are indicated by *.

| Feature Set | Precision | Recall | F1 |
|---|---|---|---|
| Word | 0.6571 | 0.6011 | 0.6279 |
| A | 0.8404 | 0.8031 | 0.8213 |
| B | 0.6167 | 0.6006 | 0.6085 |
| C | 0.7691 | 0.6726 | 0.7192 |
| BC | 0.7269 | 0.712 | 0.7194 |
| AB | 0.8368 | 0.8038 | 0.8200 |
| AC | 0.8378 | 0.8059 | 0.8216 |
| ABC | **0.8409** | **0.8066** | **0.8234*** |

Table 5: Results for word embedding features. The highest effectiveness obtained by each feature group is highlighted in bold. Statistically significant improvements ($p<0.05$) for F1 when compared with ABC are indicated by *.

| Features | Corp | Prec. | Recall | F1 |
|---|---|---|---|---|
| D | C1 | 0.7758 | **0.7392** | **0.7571** |
| | C2 | 0.7612 | 0.6926 | 0.7252 |
| | PM | **0.7776** | 0.7309 | 0.7535 |
| | WK | 0.733 | 0.6534 | 0.6909 |
| GH | C1 | 0.7868 | **0.7469** | 0.7663 |
| | C2 | 0.7847 | 0.7001 | 0.7400 |
| | PM | **0.8005** | 0.7466 | **0.7726** |
| | WK | 0.7106 | 0.6043 | 0.6532 |
| ABCD | C1 | 0.8432 | 0.8123 | **0.8275** |
| | C2 | **0.8435** | 0.8006 | 0.8215 |
| | PM | 0.8377 | **0.8126** | 0.8249 |
| | WK | 0.8409 | 0.8108 | 0.8256 |
| ABCD GH | C1 | **0.8509** | **0.8118** | **0.8309*** |
| | C2 | 0.8386 | 0.8001 | 0.8189 |
| | PM | 0.8484 | 0.8088 | 0.8281 |
| | WK | 0.8397 | 0.8063 | 0.8226 |

### 5.2 Analysis of Word Embedding Features

We study the effect of word embeddings on concept extraction to answer our RQ1 (see Section 1). To do so, we select the best combination of baseline features (ABC) and measure the effectiveness of adding semantic and lexical vectors features (groups D, G, and H). Results are reported in Table 5.

The effectiveness of the derived information extraction systems is influenced by the training corpus used to produce the embeddings. Thus, the results in Table 5 are reported with respect to the corpora; the effect training corpora have on effectiveness will be discussed in Section 5.4.

The effectiveness obtained when using the word embedding features alone[3] (group D) is comparable to that observed when using baseline semantic features (group C, Table 4). Group D includes 8 clustering features with window sizes 2 and 5. When using features of the three words preceding and following the target word with 1024 clusters (groups G and H), higher effectiveness is observed, irrespectively of the corpus (apart from WK).

Further improvements are obtained when clustering features are used in conjunction with the baseline features. The improvements in effectiveness observed when adding both D and contextual word embedding clustering features (G and H) are statistically significant compared to feature groups ABC. These results confirm those found in previous work that explored the use of word embeddings to improve effectiveness in information extraction tasks, e.g., Tang et al. (2015).

Note that we did study the effectiveness of using feature groups G and H with different number of clusters (i.e., 128, 256, 512 and 1024); however, the highest effectiveness was achieved when considering 1024 clusters. Similarly, we also experimented with different settings of word embedding's window size and dimensionality; the results of these experiments are not included in this paper for brevity[4]. The outcome of these trials was that embeddings with window size 5 usually perform better than window size 2, though not significantly; however the highest effectiveness is achieved when both sizes 2 and 5 are used. We also observed that there are no significant differences between the effectiveness of learnt models using embeddings generated with 300 dimensions as opposed to 100. However, larger embeddings are computationally more costly than smaller ones (both in terms of computer clocks and memory). Therefore, in this paper, all results were produced using embeddings of dimension 100.

### 5.3 Analysis of Sequence Features

We also study the effect of sequence features on concept extraction to answer our RQ1. For this we select the best combination of baseline and word embedding features (ABCDGH) and measure the effectiveness of adding sequence features (groups

[3] In the following, when referring to using a feature group alone, we mean using that feature group, along with the target word string.

[4] But can be found as an online appendix at https://github.com/ldevine/SeqLab.

Table 6: Results for sequence features. The highest effectiveness obtained by each feature group is highlighted in bold. Statistically significant improvements ($p<0.05$) for F1 when compared with ABC are indicated by *.

| Features | Corp | Prec. | Recall | F1 |
|---|---|---|---|---|
| J | C1 | 0.6832 | 0.6693 | 0.6762 |
| | C2 | 0.5926 | 0.6036 | 0.7012 |
| | PM | **0.7408** | **0.6701** | **0.7037** |
| | WK | 0.733 | 0.6534 | 0.6909 |
| K | C1 | **0.7646** | **0.6747** | **0.7169** |
| | C2 | 0.7241 | 0.6639 | 0.6927 |
| | PM | 0.735 | 0.6641 | 0.6978 |
| | WK | 0.7237 | 0.6609 | 0.6909 |
| ABCD GHJ | C1 | **0.8493** | **0.8136** | **0.8311** |
| | C2 | 0.8463 | 0.7968 | 0.8208 |
| | PM | 0.8475 | 0.8134 | 0.8301 |
| | WK | 0.8388 | 0.8087 | 0.8235 |
| ABCD GHK | C1 | 0.8473 | 0.8066 | **0.8265** |
| | C2 | **0.8494** | 0.7941 | 0.8208 |
| | PM | 0.8423 | 0.8061 | 0.8238 |
| | WK | 0.8399 | **0.8103** | 0.8249 |
| ABCD GHJK | C1 | 0.8488 | **0.8152** | **0.8316*** |
| | C2 | **0.8491** | 0.7959 | 0.8216 |
| | PM | 0.8472 | 0.8151 | 0.8308 |
| | WK | 0.8364 | 0.8034 | 0.8195 |
| L | C1 | 0.7601 | **0.6763** | **0.7157** |
| | C2 | 0.7311 | 0.6014 | 0.6599 |
| | PM | **0.7624** | 0.6720 | 0.7144 |
| | WK | 0.7619 | 0.6646 | 0.7099 |
| M | C1 | 0.7584 | **0.6761** | **0.7148** |
| | C2 | 0.6456 | 0.6521 | 0.6488 |
| | PM | **0.7602** | 0.6725 | 0.7137 |
| | WK | 0.6588 | 0.6424 | 0.6505 |
| ABCD GHJKL | C1 | **0.8484** | 0.8103 | 0.8289 |
| | C2 | 0.8460 | 0.7931 | 0.8187 |
| | PM | 0.8444 | **0.8147** | **0.8293*** |
| | WK | 0.8388 | 0.8024 | 0.8202 |
| ABCD GHJKM | C1 | **0.8505** | 0.8144 | **0.8320*** |
| | C2 | 0.8457 | 0.7967 | 0.8205 |
| | PM | 0.8468 | **0.8160** | 0.8311 |
| | WK | 0.8306 | 0.8060 | 0.8181 |
| ABCD GHJKLM | C1 | **0.8504** | 0.8116 | 0.8305* |
| | C2 | 0.8465 | 0.7959 | 0.8204 |
| | PM | 0.8477 | **0.8152** | **0.8311*** |
| | WK | 0.8391 | 0.8028 | 0.8205 |

J, K (sentence) and L, M (phrase)). Results are reported in Table 6.

The use of either feature groups J, K, L, M alone provide results that are comparable to the baseline semantic feature (C) or the embedding features (D), but are less effective than the use of the previous combination of features (ABCDGH).

Adding sentence features J and K separately to the remaining feature groups shows mixed results with no significant changes compared to ABCDGH. Specifically, feature group J provides small improvements across different corpora, while insignificant decrease is observed on C1 and PM with feature group K. Similar results are obtained with L and M (not reported).

However, when we combine all sentence features together (ABCDGHJK) we observe small improvements across all corpora except WK. This suggests that the results are somewhat sensitive to variation in the corpora used to learn word embeddings and sequence representations – we explore this further in the next section.

When the phrase features are added to word embedding and sentence features, small improvements are observed both over word embeddings (ABCDGH) and word embeddings with sentence features (ABCDGHJK).

In summary, sequence features provide small, additional improvements over word embedding features in the task of clinical concept extraction (when clinical and biomedical corpora are used to learn sequence representations). Given the differences between word embeddings, sentence features and phrase features, the results suggest that perhaps phrase, rather than sentence level representations should be further explored.

### 5.4 Analysis of Training Corpora

The results obtained when employing embedding features (D, G, H) and sequence features (J, K, L, M) are influenced by the corpora used to compute the embeddings (see Table 5 and 6). We therefore address our RQ2: how sensitive are the features to the training corpora?

The empirical results suggest that using a small corpus such as i2b2 (C2) to build the representations does not provide the best effectiveness, despite the test set used for evaluation contains data that is highly comparable with that in C2 (this corpus contains only i2b2's train set). However, the highest effectiveness is achieved when augmenting C2 with data from clinical corpora like Medtrack and ShARe/CLEF (C1).

The results when PubMed (PM) is used to derive the feature representations are generally lower

Table 7: Number of target tokens contained in the i2b2 test set but not in each of the word embedding training corpora.

| Corp | # Miss. Tok. | Corp | # Miss. Tok. |
|---|---|---|---|
| C1 | 196 | PM | 549 |
| C2 | 890 | WK | 1152 |

but comparable to those obtained on the larger clinical corpus (C1) and always better than those obtained on the smaller clinical corpus (C2) and the Wikipedia data (WK).

Learning word embedding and sequence features from Wikipedia, in combination with the baseline features (i.e., `ABCDGH` and `ABCDGHJKLM`), results in (small) losses of effectiveness compared to the use of baseline features only (`ABC`), despite Wikipedia being one of the largest corpora among those experimented with. We advance two hypotheses to explain this: (1) Wikipedia contains less of the tokens that appear in the i2b2 test set than any other corpora (*poor coverage*), (2) for the test tokens that do appear in Wikipedia, word embedding representations as good as those obtained from medical data cannot be constructed because of the sparsity of domain aligned data (*sparse domain data*). The first hypothesis is supported by Table 7, where we report the number of target tokens contained in the i2b2 test dataset but not in each of the word embedding training corpora. The second hypothesis is supported by a manual analysis of the embeddings from WK and compared e.g. to those reported for C1 in Table 1. Indeed, we observe that embeddings and clusters in C1 address words that are misspelled or abbreviated, a common finding in clinical text; while, the representations derived from WK miss this characteristic (see also Nothman et al. (2009)). We also observe that the predominant word senses captured by many word vectors is different between medical corpora and Wikipedia, e.g., *episodes:* {*bouts, emesis, recurrences, ...*} in C1, while *episodes:* {*sequels, airings, series, ...*} in WK.

These results can be summarised into the following observations:

- C2 does not provide adequate coverage of the target test tokens because of the limited amount of data, despite its clinical nature;
- when using medical corpora, the amount of data, rather than its format or domain, is often more important for generating representations conducive of competitive effectiveness;
- data containing biomedical content rather than clinical content can be used in place of clinical data for producing the studied feature representations without experiencing considerable loss in effectiveness. This is particularly important because large clinical datasets are expensive to compile and are often a well guarded, sensitive data source;
- if content, format and domain of the data used to derive these unsupervised features is too different from that of the target corpus requiring annotations, then the features are less likely to deliver effective concept extraction.

## 6 Conclusions and Future Work

This paper has investigated the use of unsupervised methods to generate semantic and lexical vectors, along with sequence features for improving clinical information extraction. Specifically, we studied the effectiveness of these features and their sensitivity to the corpus used to generate them. The empirical results have highlighted that:

1. word embeddings improve information extraction effectiveness over a wide set of baseline features;
2. sequence features improve results over both baseline features (significantly) and embeddings features (to a less remarkable extent);
3. the corpora used to generate the unsupervised features influence their effectiveness, and larger clinical or biomedical corpora are conducive of higher effectiveness than small clinical corpora or large generalist corpora. These observations may be of guidance to others.

This study opens up a number of directions for future work. Other approaches to create lexical vectors exits, e.g., morpheme embeddings (Luong et al., 2013), or convolutional neural nets applied at the character level (Zhang and LeCun, 2015), and their effectiveness in this context is yet to be studied. Similarly, we only investigated an initial (but novel) approach to forming sequence representations for feature generation. Given the promise expressed by this approach, more analysis is required to reach firm conclusions about the effectiveness of sequence features (both sentence and phrase), including the investigation of alternative approaches for generating these feature groups.

## References


David Arthur and Sergei Vassilvitskii. 2007. k-means++: The advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1027–1035. Society for Industrial and Applied Mathematics.

Peter F Brown, Peter V Desouza, Robert L Mercer, Vincent J Della Pietra, and Jenifer C Lai. 1992. Class-based n-gram models of natural language. *Computational linguistics*, 18(4):467–479.

Lawrence Buckingham, James M Hogan, Shlomo Geva, and Wayne Kelly. 2014. Locality-sensitive hashing for protein classification. In *Conferences in Research and Practice in Information Technology*, volume 158. Australian Computer Society, Inc.

Jeremy Buhler. 2001. Efficient large-scale sequence comparison by locality-sensitive hashing. *Bioinformatics*, 17(5):419–428.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 12:2493–2537.

Berry de Bruijn, Colin Cherry, Svetlana Kiritchenko, Joel Martin, and Xiaodan Zhu. 2011. Machine-learned solutions for three stages of clinical information extraction: the state of the art at i2b2 2010. *Journal of the American Medical Informatics Association*, 18(5):557–562.

Christopher M De Vries, Richi Nayak, Sangeetha Kutty, Shlomo Geva, and Andrea Tagarelli. 2011. Overview of the inex 2010 xml mining track: Clustering and classification of xml documents. In *Comparative evaluation of focused retrieval*, pages 363–376. Springer.

Dina Demner-Fushman, Wendy W Chapman, and Clement J McDonald. 2009. What can natural language processing do for clinical decision support? *Journal of biomedical informatics*, 42(5):760–772.

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

Thomas Hannagan, Emmanuel Dupoux, and Anne Christophe. 2011. Holographic string encoding. *Cognitive Science*, 35(1):79–118.

Siddhartha Jonnalagadda, Trevor Cohen, Stephen Wu, and Graciela Gonzalez. 2012. Enhancing clinical concept extraction with distributional semantics. *Journal of biomedical informatics*, 45(1):129–140.

Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. A convolutional neural network for modelling sentences. *arXiv preprint arXiv:1404.2188*.

Pentti Kanerva, Jan Kristofersson, and Anders Holst. 2000. Random indexing of text samples for latent semantic analysis. In *Proceedings of the 22nd annual conference of the cognitive science society*, volume 1036.

Madian Khabsa and C Lee Giles. 2015. Chemical entity extraction using crf and an ensemble of extractors. *J Cheminform*, 7(Suppl 1):S12.

Mahnoosh Kholghi, Laurianne Sitbon, Guido Zuccon, and Anthony Nguyen. 2014. Factors influencing robustness and effectiveness of conditional random fields in active learning frameworks. In *Proceedings of the 12th Australasian Data Mining Conference*, AusDM'14. Australian Computer Society.

Mahnoosh Kholghi, Laurianne Sitbon, Guido Zuccon, and Anthony Nguyen. 2015. External knowledge and query strategies in active learning: A study in clinical information extraction. In *Proceedings of the 24rd ACM International Conference on Conference on Information and Knowledge Management*, CIKM '15, New York, NY, USA. ACM.

Bevan Koopman, Guido Zuccon, Anthony Nguyen, Anton Bergheim, and Narelle Grayson. 2015. Automatic ICD-10 classification of cancers from free-text death certificates. *International Journal of Medical Informatics*, 84(11):956 – 965.

John D Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 282–289. Morgan Kaufmann Publishers Inc.

Quoc V Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. *arXiv preprint arXiv:1405.4053*.

Minh-Thang Luong, Richard Socher, and Christopher D Manning. 2013. Better word representations with recursive neural networks for morphology. *CoNLL-2013*, 104.

Andrew Kachites McCallum. 2002. Mallet: A machine learning for language toolkit. http://mallet.cs.umass.edu.

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

Frederic Morin and Yoshua Bengio. 2005. Hierarchical probabilistic neural network language model. In *Proceedings of the international workshop on artificial intelligence and statistics*, pages 246–252.

Azadeh Nikfarjam, Abeed Sarker, Karen O'Connor, Rachel Ginn, and Graciela Gonzalez. 2015. Pharmacovigilance from social media: mining adverse drug reaction mentions using sequence labeling with word embedding cluster features. *Journal of the*

*American Medical Informatics Association*, page ocu041.

Joel Nothman, Tara Murphy, and James R Curran. 2009. Analysing wikipedia and gold-standard corpora for ner training. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 612–620. Association for Computational Linguistics.

Lizhen Qu, Gabriela Ferraro, Liyuan Zhou, Weiwei Hou, Nathan Schneider, and Timothy Baldwin. 2015. Big data small data, in domain out-of domain, known word unknown word: The impact of word representation on sequence labelling tasks. *arXiv preprint arXiv:1504.05319*.

Hanna Suominen, Sanna Salanterä, Sumithra Velupillai, Wendy W Chapman, Guergana Savova, Noemie Elhadad, Sameer Pradhan, Brett R South, Danielle L Mowery, Gareth JF Jones, et al. 2013. Overview of the share/clef ehealth evaluation lab 2013. In *Information Access Evaluation. Multilinguality, Multimodality, and Visualization*, pages 212–231. Springer.

Buzhou Tang, Yonghui Wu, Min Jiang, Joshua C Denny, and Hua Xu. 2013. Recognizing and encoding discorder concepts in clinical text using machine learning and vector space model. In *Workshop of ShARe/CLEF eHealth Evaluation Lab 2013*.

Buzhou Tang, Hongxin Cao, Xiaolong Wang, Qingcai Chen, and Hua Xu. 2014. Evaluating word representation features in biomedical named entity recognition tasks. *BioMed research international*, 2014.

Buzhou Tang, Yudong Feng, Xiaolong Wang, Yonghui Wu, Yaoyun Zhang, Min Jiang, Jingqi Wang, and Hua Xu. 2015. A comparison of conditional random fields and structured support vector machines for chemical entity recognition in biomedical literature. *Journal of cheminformatics*, 7(supplement 1).

Kristina Toutanova, Dan Klein, Christopher D Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 173–180. Association for Computational Linguistics.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 384–394. Association for Computational Linguistics.

Özlem Uzuner, Brett R South, Shuying Shen, and Scott L DuVall. 2011. 2010 i2b2/va challenge on concepts, assertions, and relations in clinical text. *Journal of the American Medical Informatics Association*, 18(5):552–556.

E Voorhees and R Tong. 2011. Overview of the trec 2011 medical records track. In *Proceedings of TREC*, volume 4.

Dominic Widdows and Trevor Cohen. 2014. Reasoning with vectors: A continuous model for fast robust inference. *Logic Journal of IGPL*, pages 141–173.

Xiang Zhang and Yann LeCun. 2015. Text understanding from scratch. *arXiv preprint arXiv:1502.01710*.