Extracting chemical-disease associations from the biomedical literature
R214: Main Practical

Jan Ondras (jo356)
Trinity College

April 23, 2018

Abstract

In this report I analyse the strength of associations between chemical and disease co-mentions in the biomedical literature. First, I tune and train the named entity recognition (NER) model based on Conditional Random Fields to tag named entities (NEs) in text as chemicals or diseases. Next, using approximate string marching I build a system to ground NEs to Medical Subject Headings (MeSH) concepts. Lastly, the NER model and the grounding system are applied to the corpus of *PubMed* abstracts on chemically-induced disorders. I then analyse the occurrences of chemical-disease pairs (CDPs) using various co-occurrence measures and also investigate the similarity between the rankings of CDPs produced by these measures. I further classify the CDPs according to the type of the relation between the chemical and disease, consulting with the physician MUDr Maria Kleinova.

My results show that the chemical *"levodopa"* and the disease *"abnormal movements"* co-occur most frequently according to Symmetric Conditional Probability (SCP) and Dice Coefficient (DC) measures and for each co-occurrence measure this CDP also appears in the top 10 ranked CDPs. My investigation further indicates that the Simple Co-occurrence Count (SCC) is unlikely to be useful for discovering new chemical-disease associations whereas the Normalised Point-wise Mutual Information (NPMI) is promising for this task. Also, the ranking of CDPs by SCC measure is most dissimilar to rankings by other measures. Regarding the type of the relation between chemicals and diseases, the SCC measure seems to be best suited for identification of CDPs where the chemical *induces* the disease, while the NPMI measure for extraction of CDPs with *not very well known* or possibly *unknown* relations.

Contents

1 Introduction

Chemical-disease relations (CDRs) play an important role in healthcare and biomedical research [1]. Extracting and understanding these relations is thus of great importance for various biomedical tasks such as development of new drugs or therapies. Some systems such as the Comparative Toxicogenomics Database [2] address this task by manual annotation of the CDRs which generates accurate knowledge but is very costly and difficult to keep up-to-date given the rapid accumulation of the biomedical literature. Therefore, automatic biomedical information extraction systems based on natural language processing methods present a great alternative. For instance, Gu et. al [3] apply a machine learning approach based on Mallet MaxEnt classifier to extract the CDRs. They used the *BioCreative V CDR* dataset [4] and achieved promising F-scores of 60.4% and 58.3% on the development and test set respectively, using the ground truth annotations of named entities. Another work [5], by Xu et. al, extracts the CDRs using the system that consists of: named entity (NE) recognition module based on Conditional Random Fields, NE normalisation module based on Vector Space Models, and a relation extraction module that classifies candidate chemical-disease pairs using Support Vector Machines.

In this project, I perform the named entity recognition and normalisation (grounding), similarly to [5]. However, instead of the supervised approach to identification of chemical-disease associations, I use an unsupervised method based on various co-occurrence measures.

The rest of the report is structured as follows. Section 2 describes the named entity recognition of disease and chemical entities and Section 3 explains the subsequent grounding technique. The extracted chemical-disease co-occurrences are presented and analysed in Section 4. Section 5 concludes the report and refers to possible directions for further improvements of the presented method.

2 Named entity recognition

The goal of the named entity recognition (NER) is to find and classify named entities in text into a set of predefined categories. For example, a correctly tagged sentence with the set of categories $\{Chemical, Disease\}$ is:

There was only one case of Disease NE[*dementia*] *possibly due to* Chemical NE[*cimetidine*] .

NER is a long-standing problem with a wide range of application areas. Starting with the extraction from journalistic articles in the 1990s, the attention steered to military reports and later to informal texts such as weblogs. Over the last two decades the focus turned to biomedical domain.

In general, there are three kinds of approaches to NER proposed in the literature: rule-based [6, 7], machine learning based [8, 9] and hybrid solutions [10, 11]. While the rule-based systems benefit from better interpretability, most of the state-of-the-art results for NER problems are based on machine learning techniques. Besides the traditional sequence labeling models such as Hidden Markov Models or Conditional Random Fields (CRF) [12] that rely on hand-crafted features, the non-linear neural network models of various types and architectures have recently become popular. For instance, Bidirectional Long Short-Term Memory (BLSTM) networks [13], and also hybrid methods combining CRF and BLSTM [14, 15] or even CRF, BLSTM and Convolutional Neural Networks [16]. For a reference, the state-of-the-art NER systems for English achieve near-human performance and F-scores as high as 91.21% [16]. Lastly, it is important to note that even the best NER systems designed for one domain do not usually perform well on other domains [17], which implies that the NER models should be trained and tuned on data from the target domain.

In this project I developed the NER model based on Conditional Random Fields using the *CRFsuite* toolbox [18], in order to identify chemical and disease named entities in texts.

2.1 Dataset

I trained, tuned and tested the NER model on the *BioCreative V CDR* dataset [4] with details shown in Tab. 1. This text corpus consists of 1500 *PubMed* articles with 4409 annotated chemicals, 5818 diseases and 3116 chemically-induced disease (CID) relations.

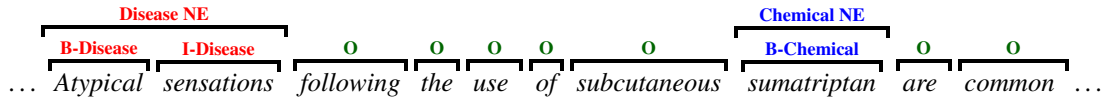
Table 1: *BioCreative V CDR* dataset [4] statistics used for development of NER CRF-based model. CID stands for chemically-induced disease.

			Chemical		Disease		
	# articles	# sentences	# entities	# mentions	# entities	# mentions	# CID relations
Training set	500	4560	1467	5203	1965	4182	1038
Development set	500	4579	1507	5347	1865	4244	1012
Test set	500	4797	1435	5385	1988	4424	1066

In this dataset each text item is tagged with one of the following labels using the IOB2 notation in accordance with the chemical and disease Medical Subject Headings (MeSH) concepts.

- **B-Chemical** – beginning item of chemical NE.
- **I-Chemical** – inside item of chemical NE.
- **B-Disease** – beginning item of disease NE.
- **I-Disease** – inside item of disease NE.
- **O** – other item not belonging to chemical nor disease NE.

An excerpt from the training set is shown below (a sequence of items is assigned with a sequence of item labels):



2.2 Procedure

Firstly, I extracted the following features for each item, using the *CRFsuite* toolbox [18].

- **word[0]** – word of the current item.
- **word[-1]** – word of the previous item.
- **word[1]** – word of the next item.
- **lemma[0]** – lemma of the current item.
- **soundex[0]** – phonetic encoding of the current item (items indexed by sound as pronounced in English).
- **pos[0]** – part-of-speech tag of the current item.
- **chunk[0]** – beginning/inside/other phrase tag of the current item.

The importance of each feature was then investigated on the development set by means of feature ablation experiments (Sec. 2.3). Next, I searched for the optimal L2 regularisation parameter c_2 of the L-BFGS optimiser [19] used by CRFsuite (Sec. 2.4). Finally, I compared the *default model* with the *tuned model* on the test set (Sec. 2.5).

2.3 Feature ablation

The performance of the NER model on the development set when various features were excluded is shown in Tables 2, 3 and 4 in terms of precision, recall and F1 score respectively. Also, I provide a graphical illustration of these results in Appendix A (Fig. 6).

We can observe that the **precision** for item classes *I-Chemical*, *B-Disease* and *I-Disease* as well as for *Overall* degrades considerably when the feature *word[-1]* is ablated. This shows that the information about previous word is important for correct tagging of the current word which could be expected, especially in the case of inside item classes *I-Chemical* and *I-Disease*. Differently, maximum *Overall* (as well as *B-Chemical*, *I-Chemical*

and *I-Disease*) precision is achieved when the *pos[0]* feature is excluded. This indicates that the part-of-speech information is not very relevant for high precision predictions of the two NEs of interest and that it may cause confusion. However, the *pos[0]* feature seems to be informative for precise tagging of other items (with label *O*) not belonging to chemical nor disease NEs.

The results for **recall** also show the importance of the *word[-1]* feature for correct identification of inside item classes *I-Chemical* and *I-Disease* as well as the class *O*. We can notice that the ablation of *pos[0]* feature has directly opposite effect on recall than on precision, namely, the exclusion of part-of-speech information results in the lowest recall for *B-Chemical* class (considerably smaller) as well as *Overall* and in the highest recall for class *O*. This implies that in the problem investigated the *pos[0]* feature presents a trade-off between precision and recall. Overall and also for the classes *B-Chemical*, *B-Disease* and *I-Disease* the best recall is obtained when all features are used which suggests that the ablation of features results in more conservative predictions.

To better evaluate and compare the overall system performance, I used the **F1 score** that combines precision and recall measures. As it can be seen from Tab. 4, the *word[-1]* feature carries important information which is in accordance with the precision and recall results. We can further observe the afore-mentioned trade-off behaviour for the *pos[0]* feature. Considering its F1 scores it seems that its overall importance is neither crucial nor negligible. Most importantly, the F1 scores reveal that the ablation of *chunk[0]* feature yields the best overall performance suggesting that the information about the position of an item in a phrase and the type of the phrase cause confusion rather than improvement in the task of chemical and disease NER. In particular, the exclusion of *chunk[0]* feature resulted in 0.1170% improvement in F1 score over the default set of all features. For this reason I further used the feature set without the *chunk[0]* features.

Table 2: Feature ablation experiments: **precision** for each item class and overall (macro-average). Evaluated on development set. *Red:* maximum per item class. *Blue:* minimum per item class.

Ablated feature	B-Chemical	I-Chemical	B-Disease	I-Disease	O	Overall
–	0.9174	0.7543	0.8404	0.7406	**0.9560**	0.841760
word[0]	0.9062	0.7585	0.8403	0.7438	0.9546	0.840674
word[-1]	0.9310	**0.7115**	**0.8259**	**0.7111**	0.9532	**0.826536**
word[1]	0.9423	0.7489	0.8373	0.7451	0.9516	0.845053
lemma[0]	0.9056	0.7586	**0.8420**	0.7470	0.9540	0.841452
soundex[0]	**0.9011**	0.7614	0.8385	0.7507	0.9532	0.840986
pos[0]	**0.9501**	**0.7954**	0.8402	**0.7643**	**0.9498**	**0.859964**
chunk[0]	0.9215	0.7705	0.8395	0.7491	0.9557	0.847258

Table 3: Feature ablation experiments: **recall** for each item class and overall (macro-average). Evaluated on development set. *Red:* maximum per item class. *Blue:* minimum per item class.

Ablated feature	B-Chemical	I-Chemical	B-Disease	I-Disease	O	Overall
–	**0.6665**	0.5973	**0.6006**	**0.6026**	0.9888	**0.691157**
word[0]	0.6592	0.5967	0.5763	0.6018	0.9886	0.684549
word[-1]	0.6304	**0.5784**	0.5836	**0.5809**	**0.9882**	0.672312
word[1]	0.6082	0.5801	0.5846	0.5846	0.9899	0.669471
lemma[0]	0.6566	0.5950	0.5664	0.6004	0.9888	0.681448
soundex[0]	0.6527	0.5932	**0.5568**	0.5956	0.9887	0.677405
pos[0]	**0.5697**	0.5915	0.5799	0.6022	**0.9910**	**0.666851**
chunk[0]	0.6652	**0.6013**	0.5978	0.5949	0.9894	0.689717

Table 4: Feature ablation experiments: **F1 score** for each item class and overall (macro-average). Evaluated on development set. *Red:* maximum per item class. *Blue:* minimum per item class.

Ablated feature	B-Chemical	I-Chemical	B-Disease	I-Disease	O	Overall
–	0.7721	0.6667	**0.7006**	0.6645	0.9721	0.755196
word[0]	0.7632	0.6679	0.6837	0.6653	0.9713	0.750310
word[-1]	0.7518	**0.6381**	0.6840	**0.6395**	0.9704	**0.736726**
word[1]	0.7393	0.6538	0.6885	0.6552	0.9704	0.741412
lemma[0]	0.7613	0.6669	0.6773	0.6657	0.9711	0.748454
soundex[0]	0.7570	0.6669	**0.6692**	0.6642	0.9706	0.745597
pos[0]	**0.7123**	**0.6785**	0.6862	**0.6736**	**0.9700**	0.744105
chunk[0]	**0.7727**	0.6754	0.6983	0.6631	**0.9723**	**0.756366**

2.4 Hyperparameter tuning

For the above-determined best-performing feature set (with *chunk[0]* feature excluded) I tuned the L2 regularisation parameter c_2 of the L-BFGS optimiser [19] used by *CRFsuite*. Firstly, I searched over the range $c_2 = 10^i$ where $i \in \{-5, -4, -3, \ldots, 3\}$ and consequently, with finer steps around the appearing maximum with $i \in \{-1.\overline{7}, -1.\overline{5}, -1.\overline{3}, \ldots, -0.\overline{2}\}$. The results in terms of the F1 score evaluated on the development set are shown in Fig. 1 and a more detailed summary for the first coarser search is provided in Tab. 5. We can see that the default value of $c_2 = 1$ was suboptimal and that the best setting in this case is $c_2 = 0.1$ which brings an improvement in F1 score by 1.9169%.

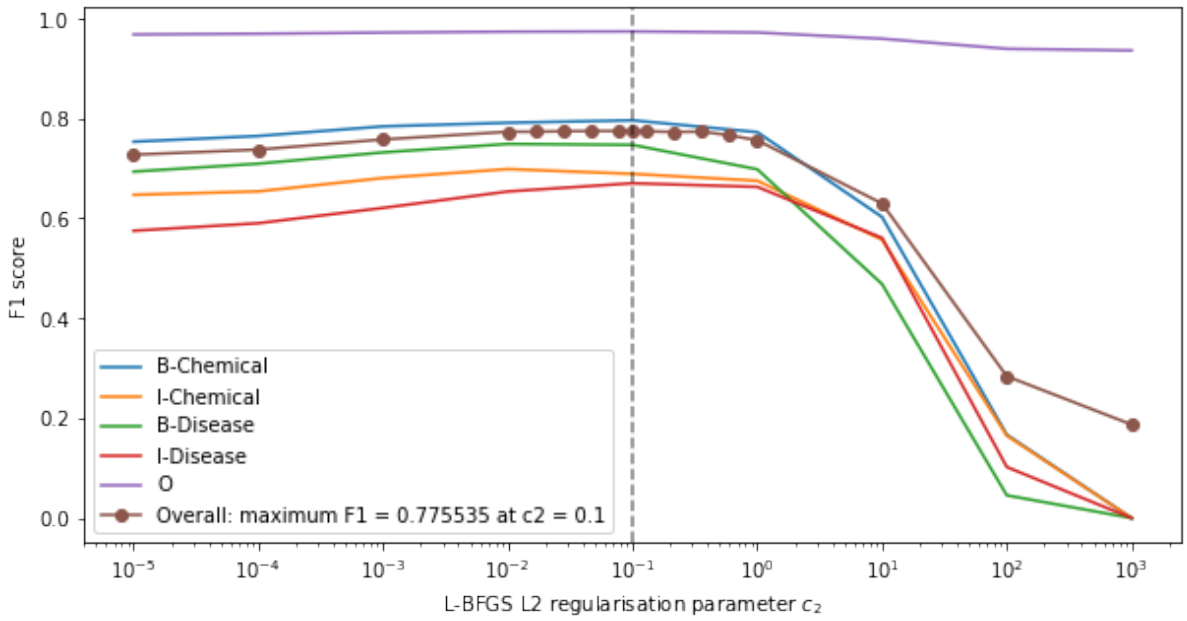


Figure 1: Hyperparameter tuning of the L-BFGS L2 regularisation parameter c_2 on the development set, using the best feature set. The default parameter value was $c_2 = 1$.

Table 5: Hyperparameter tuning of the L-BFGS L2 regularisation parameter c_2: **F1 score** for each item class and overall (macro-average). Evaluated on development set. The default parameter value was $c_2 = 1$.

c_2	B-Chemical	I-Chemical	B-Disease	I-Disease	O	Overall
10^{-5}	0.7536	0.6472	0.6937	0.5753	0.9683	0.727621
10^{-4}	0.7650	0.6541	0.7096	0.5904	0.9698	0.737764
10^{-3}	0.7843	0.6809	0.7322	0.6210	0.9721	0.758100
10^{-2}	0.7915	0.6988	0.7491	0.6539	0.9739	0.773423
$\mathbf{10^{-1}}$	0.7964	0.6892	0.7472	0.6704	0.9744	**0.775535**
10^{0}	0.7727	0.6754	0.6983	0.6631	0.9723	0.756366
10^{1}	0.6028	0.5573	0.4685	0.5609	0.9598	0.629842
10^{2}	0.1669	0.1651	0.0455	0.1022	0.9396	0.283873
10^{3}	0.0000	0.0000	0.0000	0.0000	0.9363	0.187263

2.5 Model comparison

In this section I compare the performance of the following two NER models.

- **Default model**
  - all 7 features are used
  - L-BFGS L2 regularisation parameter is set to default value $c_2 = 1$
- **Tuned model**
  - feature *chunk[0]* is ablated (according to Sec. 2.3)
  - L-BFGS L2 regularisation parameter is set to the optimal value $c_2 = 0.1$ (according to Sec. 2.4)

As shown in the previous sections, the *tuned model* achieved 2.0339%[1] overall improvement in F1 score on the development set. In what follows I further compare the two models on the test set. The results in terms of precision, recall and F1 score for each item class and overall are summarised in Tab. 6. We can see that the *default model* achieved better precision for both disease item classes *B-Disease* and *I-Disease*. This suggest that the *tuned model* did not pick up the ability of high-precision disease predictions from the *default model* very well and it might be a good starting point for further improvement. Looking at the recall measure the *tuned model* is inferior only for the *I-Chemical* class. Overall, we can conclude that the *tuned model* outperforms the *default model* in all three measures. Specifically, it improves the F1 score by 1.8590% which is slightly less but still close to the improvement of 2.0339% on the development set. This indicates that the *tuned model* generalises well.

Table 6: Comparison of the *default* and *tuned* model on the test set in terms of precision, recall and F1 score for each item class and overall (macro-average).

Measure	Model	B-Chemical	I-Chemical	B-Disease	I-Disease	O	Overall
Precision	Default	0.9082	0.7442	**0.8214**	**0.7316**	0.9576	0.832590
	Tuned	**0.9310**	**0.7815**	0.8137	0.7219	**0.9626**	**0.842159**
Recall	Default	0.6464	**0.6093**	0.5893	0.6175	0.9882	0.690133
	Tuned	**0.6594**	0.5977	**0.6684**	**0.6573**	**0.9886**	**0.714282**
F1 score	Default	0.7553	0.6700	0.6862	0.6697	0.9727	0.750779
	Tuned	**0.7720**	**0.6773**	**0.7339**	**0.6881**	**0.9754**	**0.769369**

In addition, for each model I constructed the associated confusion matrix, as shown in Fig. 2. Besides confirming the recall results (on matrix diagonal), this allows us to see which item classes are most frequently confused. In

[1] 2.0339% = 0.1170% (by feature ablation) + 1.9169% (by hyperparameter tuning)

particular, we can see that for both models each item class (except *O*) is considerably confused with the class *O* and this confusion is dominant when compared to other confusions. Therefore, further model improvements could focus on incorporation of features that would allow better discriminability between the class *O* and all the other item classes.

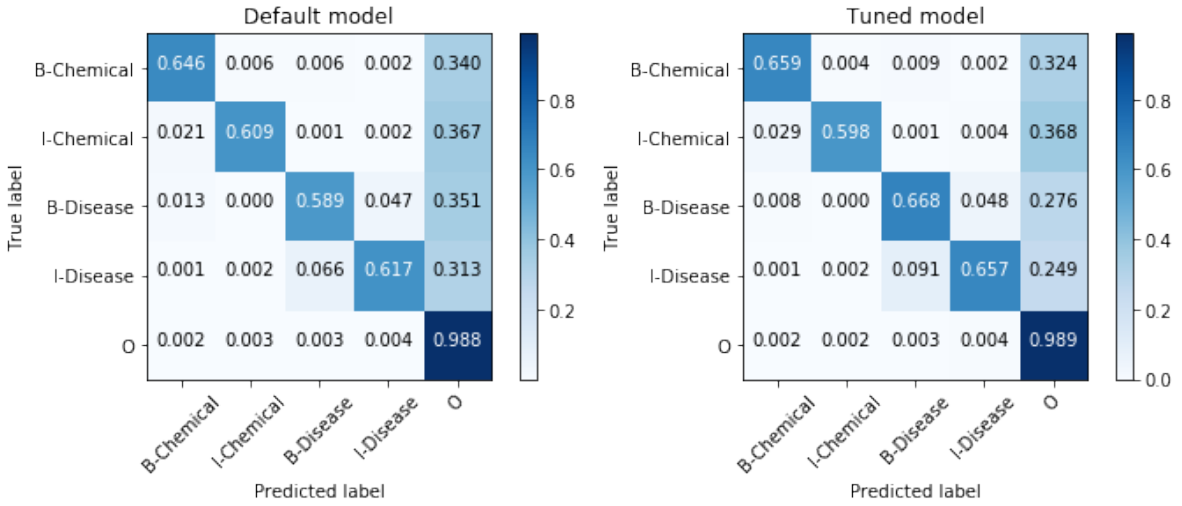


Figure 2: Normalised confusion matrices from testing of the named entity recognition CRF-based models on the test set. Left: *default model*. Right: *tuned model*.

Comparison of the *tuned model* with the state-of-the-art CRF models such as [20] that achieved F1 score of 84.04% on CoNLL03 English NER task or [21] with F1 score of 80.05% on Twitter NER task, suggests that the *tuned model* could be further improved, for instance, by experimenting with another features and/or by training on a larger dataset.

3 Grounding of named entities

This section presents a procedure used to ground the recognised named entities to Medical Subject Headings (MeSH) concepts of chemicals and diseases[2]. All 2677 MeSH entries are in a form of triples $< \texttt{Name}, \texttt{Type}, \texttt{ID} >$ where $\texttt{Type} \in \{Chemical, Disease\}$ and `ID` is a unique chemical/disease concept identifier so that multiple entries with different `Name` field can share the same `ID` value.
For example:

$$< \texttt{Parkinson's disease}, \texttt{Disease}, \texttt{D010300} >$$
$$< \texttt{Parkinson's Disease}, \texttt{Disease}, \texttt{D010300} >$$
$$< \texttt{PD}, \texttt{Disease}, \texttt{D010300} >$$

The task is to match every recognised NE string with the corresponding MeSH concept, if such a concept exists in the MeSH dataset. For instance, the chemical NE *"Amphotericin B"* is assigned with the MeSH concept ID *"D000666"* by comparing the NE string *"Amphotericin B"* with the MeSH entry name *"amphotericin B"*. I addressed this problem using the simple approximate string matching (SASM) as well as fuzzy approximate string matching (FASM).

3.1 Simple approximate string matching (SASM)

The key idea behind my procedure is to *abstract* both the MeSH entry names and NE strings to a common ground. I achieve this by omission of spaces and removal of special characters from both kinds of strings as well as by conversion to lower case letters.

[2] Available at: http://131.111.179.130/static/bmip/CDR_MeSH.tsv

The diagram of the proposed grounding system is shown in Fig. 3. As a preprocessing step I construct the *Grounding dictionary* that maps the abstracted MeSH entry name `Name*` to the rest of its MeSH entry $<$ `Type, ID` $>$. In the matching phase I abstract the NE string `Str` to string `Str*` using the same deterministic abstraction technique as in the preprocessing stage and then check whether `Str*` is a key in the *Grounding dictionary*. If yes, and the dictionary value $<$ `Type, ID` $>$ is retrieved, the NE string `Str` is matched with the MeSH concept represented by `ID`. Otherwise, the NE string `Str` is considered to have no corresponding MeSH concept.

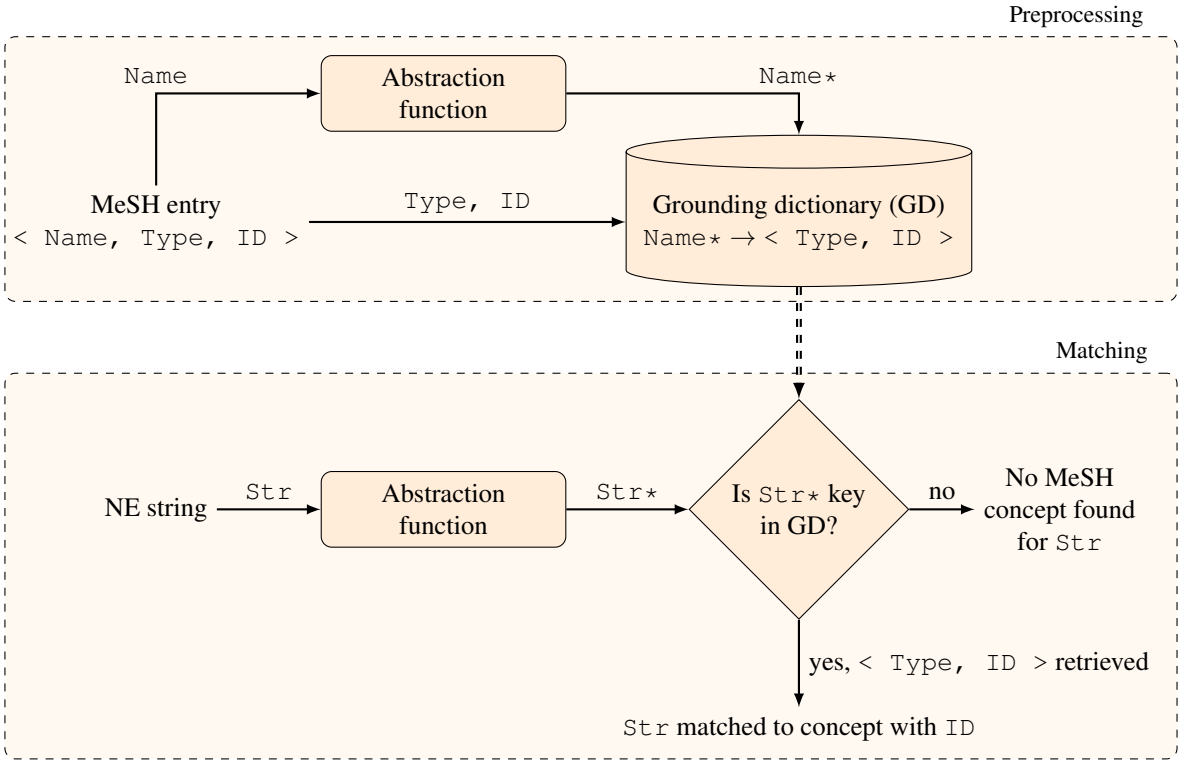


Figure 3: Grounding system based on simple approximate string matching.

To design an appropriate abstraction technique I examined the MeSH entries as well as the NE strings. In summary, I found variations in word concatenation, capitalisation and presence/absence of some special characters. I thus experimented with the following five abstraction methods that can be grouped into two broader categories.

Exact – the abstraction function is an identity function so that the MeSH entry names are themselves keys to the *Grounding dictionary*. In this case there are two ways to construct NE strings:

- **E1** – NE items are concatenated *without* a space between them and the constructed NE string is then passed through the identity abstraction function.
- **E2** – NE items are concatenated *with* a space between them and the constructed NE string is then passed through the identity abstraction function.

Approximate – the abstraction function is *not* an identity function and applies one of the following operations. Since each of the below-defined methods removes spaces it does not matter how the NE items are concatenated.

- **A1** – all spaces are removed.

$$\textit{renal failure} \xrightarrow{\text{Abstraction}} \textit{renalfailure}$$

- **A2** – all spaces are removed and the string is converted to lower case.

$$\textit{Atrial Fibrillation} \xrightarrow{\text{Abstraction}} \textit{atrialfibrillation}$$

- **A3** – all spaces and all special characters -',/*+.!?"@#$%^&*()_={}[];\:<> are removed and the string is converted to lower case.

$$\textit{N(6)-cyclopentyl adenosine} \xrightarrow{\text{Abstraction}} \textit{n6cyclopentyladenosine}$$

To compare the above-described methods I evaluated my grounding system on the train, development and test partition of the *BioCreative V CDR* dataset [4] (described in Sec. 2). To construct the NE strings I used the ground truth item labels since the NER model was already evaluated in Sec. 2 and I did not want to confuse errors caused by incorrect NER with those caused by incorrect grounding of NEs. All the compared methods were designed to be very conservative so that given a NE string that does not truly correspond to any MeSH entry it is almost impossible for it to be incorrectly matched with some MeSH entry. For this reason, the number of matches can be used for evaluation and the higher the count is the better the method is.

The results in terms of the number of NEs that were matched with MeSH concepts are shown in Tab. 7. We can see a substantial improvement from method *E1* to *E2* which shows that in the case of *"exact"* matching (identity abstraction function) a NE string constructed using spaces between its NE items can better match a MeSH entry name. In other words, if *"exact"* matching had to be used the NE items should be concatenated with spaces between them. As expected, the *"approximate"* methods (*A1*, *A2*, *A3*) outperform the *"exact"* ones (*E1*, *E2*). We can also notice that the three *"approximate"* methods generate the same number of matches on the training set which highlights the importance of evaluation on larger sample of data (inclusion of development and test sets). In particular, I checked that the equal match counts of the *"approximate"* methods on the training set are caused by the fact that the MeSH dictionary already handles some of the lower/upper case variations and omission of some special characters. However, it is not sufficient to rely on this as it can be seen from the results for the development and test sets (improvements from *A1* to *A2* and from *A2* to *A3*). Also, it is interesting to note that the improvement from *A2* to *A3* on the test set was primarily caused by removal of the dash character, namely, it accounted for 31 out of 33 new matches. Overall, the *"approximate"* method *A3* achieved the best performance and was thus chosen for the final grounding system later used in Sec. 4.

Table 7: Comparison of abstraction methods used by the grounding system based on simple approximate string matching. The counts involve both chemical and disease named entities (NEs).

	# matched NEs		
Abstraction method	Training set	Development set	Test set
E1	7113	5037	5014
E2	8779	5682	5682
A1	9309	5848	5845
A2	9309	6085	6101
A3	**9309**	**6098**	**6134**
Total # NEs	9385	9591	9809

To further improve this SASM grounding system the following points can be considered and investigated.

- Even though the MeSH dictionary contains both singular and plural entry names for some concepts, it may not hold for all concepts as it happened to be the case with the capitalisation on the training set mentioned above. This could be addressed by lemmatisation of the words within the abstraction function.
- The proposed simple matching technique does not handle variations in word ordering within strings which could be solved by an abstraction function that splits the input string into an unordered set of words and so the key into the *Grounding dictionary* would be this set.
- The SASM does not allow for a more specific NE string to be matched with a more general MeSH entry name (that carries a relevant substring of the NE string), for instance, *"impaired memory performance"* would not be matched with *"impaired memory"* (this example was chosen from the test set). To tackle this issue some form of concept taxonomy would have to be used.

3.2 Fuzzy approximate string matching (FASM)

An alternative approach is to use fuzzy approximate string matching (FASM). Given a NE string this method first calculates matching scores between the NE string and *all* MeSH entry names, using a distance metric to measure the similarity between the two strings. The MeSH entry with the highest score s is then retrieved as a match.

I experimented with the FASM method using the Python package *fuzzywuzzy*[3] that uses the Levenshtein distance [22] as the string similarity measure and provides matching scores $s = 0, 1, \ldots, 100$ where 0 corresponds to the worst and 100 to the best match. I examined this approach on the test set as it seems to contain more challenging NEs for grounding, as shown by experiments in Sec. 3.1. In this case I concatenated NE items (into NE strings) with spaces between them since such a method resulted in more relevant matches with MESH concepts (Tab. 7).

Sample matches with associated scores are shown in Tab. 8. We can see that the FASM can naturally handle:

- change in word ordering (line 8),
- spelling differences (line 10),
- singular/plural forms (line 11),
- omission of special characters (line 12),
- and capitalisation (line 13),

resulting in high scores. However, incorrect matches do not necessarily have low scores as on lines 1, 2, and 3 but also higher scores which makes the filtering of matches by the score value challenging. For example, the match on line 4 is wrong semantically but it obtained a relatively high score of 87. Similarly, the line 7 shows an incorrect match (having opposite meaning) with even higher score of 95. An interesting phenomenon can be seen on line 6 where the high-scoring match is correct most likely by a chance since the concept abbreviation happened to be similar to the concept name. Another point that should be considered in a more advanced grounding system is whether a NE of a specific chemical/disease should be matched with a more general chemical/disease concept (line 9) and vice-versa, whether a NE of a broader chemical/disease should be matched with a more specific chemical/disease concept (line 5). I believe that the former case (line 9) should be accepted since NEs and concepts exhibit the IS-A relationship and so the latter case (line 5) should not be considered as a correct match. However, the FASM treats both these cases similarly and assigns them with high scores.

Table 8: Sample matches: NE strings grounded to MeSH concepts using fuzzy approximate string matching.

		Matched MeSH entry			
	NE string	Name	Type	ID	Score
	...	...	...	...	...
1	Pb	VP	Chemical	D005047	50
2	Azotemia	anemia	Disease	D000740	71
3	biotin	nicotine	Chemical	D009538	71
4	hypokalemia	hypocalcemia	Disease	D006996	87
5	renal disease	cystic renal diseases	Disease	D052177	90
6	PAP	papaverine	Chemical	D010208	90
7	Hodgkin lymphoma	non-Hodgkin's lymphoma	Disease	D008228	95
8	learning and memory impairments	impairments in learning and memory	Disease	D007859	95
9	hepatitis C virus infection	hepatitis virus infection	Disease	D006525	96
10	diethylstilbesterol	diethylstilbestrol	Chemical	D004054	97
11	corticosteroid	corticosteroids	Chemical	D000305	97
12	prolonged QT syndrome	prolonged Q-T syndrome	Disease	D008133	98
13	Zidovudine	zidovudine	Chemical	D015215	100
	...	...	...	...	...

[3] https://github.com/seatgeek/fuzzywuzzy

Even though the FASM can find a best match (from the concept dictionary) for every NE, this is not always desirable since it is very likely that there will be some NE strings without any relevant MeSH entry. Therefore, it is crucial to set an appropriate threshold score s_T such that only matches with $s \geq s_T$ are accepted. Otherwise, incorrect matches (with low scores) would be generated. However, as noted above, finding a suitable value for s_T is tricky since even high-scoring matches may be incorrect. One possible way to estimate this threshold s_T would be to generate best matches for all NEs and manually label all of them as either correct or incorrect. Then the optimal score threshold s_T could be determined by minimising the distance $d = \sqrt{(1 - s_n)^2 + (1 - s_p)^2}$ in the ROC curve plot [23], where s_n and s_p denote sensitivity and specificity of the matchings respectively. This estimation should be performed on a training set and then applied to a test set.

To better see what values of matching scores were produced, I plotted the distribution of matches by their scores in Fig. 4 and the associated cumulative distribution in Fig. 5. The majority of matches scored 100, namely 5936 (out of 9809) which is less than the 6134 matched NEs generated by the best SASM method *A3* (Tab. 7). Therefore, it is necessary to also consider lower-scoring matches. We can further observe a step in the cumulative distribution: 8743 matches with score ≥ 90 and 6360 matches with score ≥ 91. This allows us to hypothesise that the threshold value $s_T = 90$ would probably result in many incorrect matches, while the value $s_T = 91$ might be appropriate and is likely to provide an improvement over the grounding system from Sec. 3.1. However, in the case of the FASM we cannot be certain about the correctness of the retrieved matches and thus neither about the improvement.

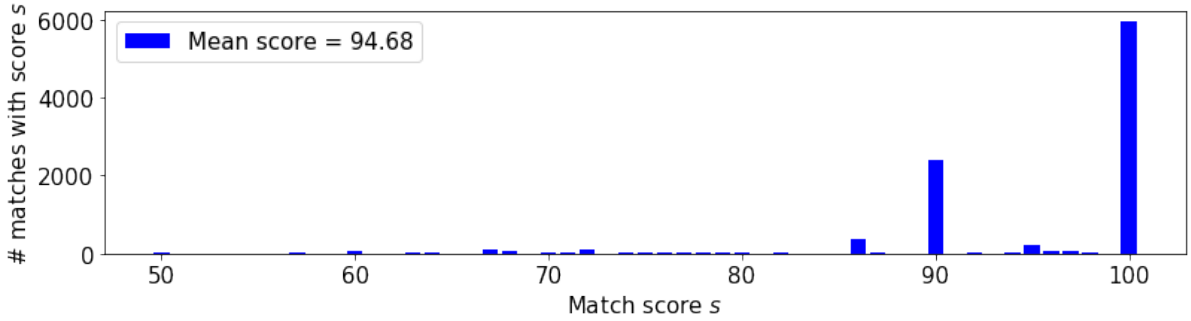


Figure 4: Distribution of matches according to their scores. 5936 matches with score 100 and 2383 matches with score 90.

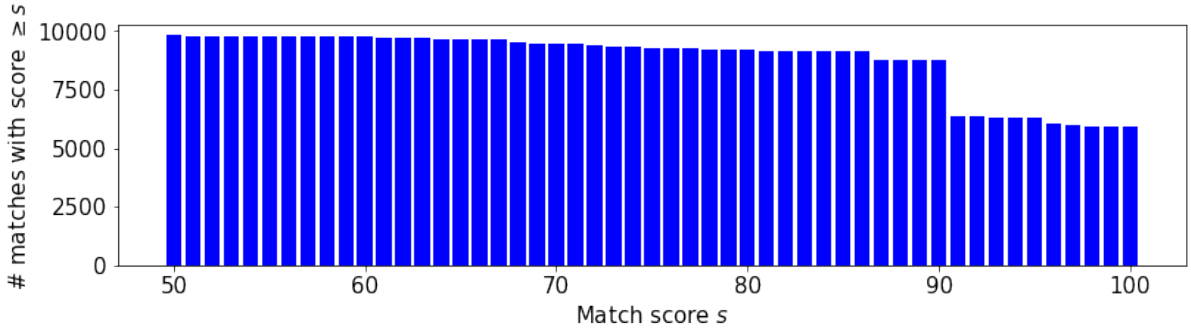


Figure 5: Cumulative distribution of matches: # matches with score s and higher. 8743 matches with score ≥ 90 and 6360 matches with score ≥ 91.

To summarise, the FASM allows more flexible grounding of NEs to MeSH concepts when compared to the SASM. However, it relies on the threshold parameter s_T whose proper setting is challenging and it would probably involve costly manual inspection and annotation of matches. Moreover, the time complexity of the FASM is its significant downside as it compares the given NE string with *all* ($n = 2677$) MeSH entry names resulting in the time

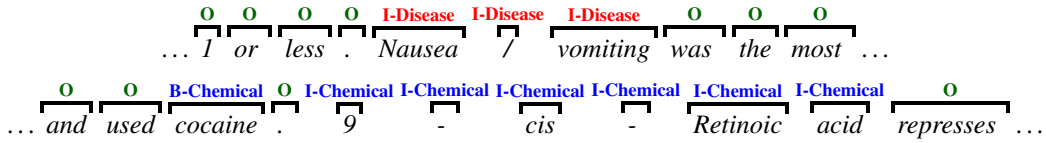
complexity $O(n)$ for each NE string. In contrast to the FASM, the SASM method just keys the *Grounding dictionary* which takes $O(1)$ time for each NE string.[4] This suggests that when large texts of NEs need to be grounded, the FASM is computationally efficient only if the dictionary of concepts (in this case MeSH) is small. In particular, the running times of FASM and SASM on the test set were 27 minutes and 0.08 seconds respectively[5]. Considering the size of the target *PubMed* dataset, it would take around 16 days instead of 21 minutes to process such amount of data. This was the reason why I decided not to perform the final evaluation of the FASM-based grounding system on the target *PubMed* dataset.

4 Chemical-disease co-occurrences

In this section I apply the *tuned* NER model (Sec. 2.5) and the SASM-based grounding system (Sec. 3.1) to the *PubMed* abstract texts on chemically-induced diseases[6]. I then analyse the co-occurrences of chemical concepts with disease concepts.

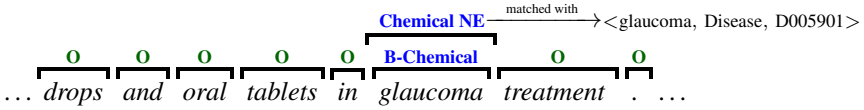
4.1 Procedure

Firstly, I tagged the *PubMed* dataset using the NER model. Looking at the NE item tags I observed that there were some NEs without a beginning item tag, namely, I found 452 such NEs. For instance:



Further investigation showed that in all such cases a dot preceded the NE and the constructed NEs (even without the beginning item) seemed to be meaningful which suggests that the dot caused the confusion and incorrect item tag assignment and also indicates that such NEs should be accepted. Therefore, I adjusted my technique to treat such a sequence of NE items (without a beginning item) as a proper NE. Also, for each NE I checked whether the category (chemical/disease) of NE items within the NE is consistent. This was always the case.

Next, I performed the SASM-based grounding of the recognised NEs to MeSH concepts. I noticed that the NE class (chemical/disease) sometimes disagreed with the `Type` field of the matched MeSH entry. In total, 5,940,715 (out of 9,105,005) NEs were grounded to MeSH concepts and in 4,261 cases there was a mismatch between the NE class and the `Type` of the matched MeSH concept. For example:



We can be almost sure that this mismatch is caused by incorrect NE tag assignment made by the NER model since the MeSH entries are considered to be true and well checked. So I set the matched concept category according to the `Type` field instead of the NE tag given by NER model, which improves the overall system correctness.

Lastly, for every chemical c and disease d I calculated the total number N_c of sentences where the chemical c occurred, the total number N_d of sentences where the disease d occurred, and the total number N_{c-d} of sentences where the chemical c and the disease d co-occurred. All these counts were computed on a per-sentence basis so that multiple occurrences/co-occurrences within a single sentence contributed to total counts only by one. The sentences in *PubMed* abstracts are separated by empty lines making the sentence segmentation straightforward. To lower the memory requirements, all the counts were stored in a form of dictionaries so that at every point during the processing only the entries with non-zero counts were kept. More specifically, these counts were computed while sequentially processing the recognised NE items. Once the whole NE was parsed and a match

[4] Of course, the SASM involves the preprocessing operation to create the *Grounding dictionary* which takes $O(n)$ time. However, it is done only once, not for each NE string.

[5] The measurements were made on my laptop with quad-core 2.2GHz Intel i7 CPU and 8GB RAM.

[6] Available at http://131.111.179.130/static/bmip/pubmed-abstracts.conll.gz, containing 10,573,978 sentences which is considerably more than the *BioCreative V CDR* dataset (Tab. 1).

with a MeSH concept was found, the temporary within-sentence counts were updated. After the whole sentence was parsed, the total counts were updated based on the temporary within-sentence counts. In total, 46,137 unique chemical-disease pairs (CDPs) were extracted.

4.2 Co-occurrence measures

To evaluate the associations between chemicals and diseases I used the following co-occurrence measures where $P(c)$ is the probability that the chemical c occurs in a sentence, $P(d)$ is the probability that the disease d occurs in a sentence, $P(c,d)$ is the probability that the chemical c co-occurs with the disease d in a sentence, and $N = 10,573,978$ is the total number of sentences in the employed *PubMed* dataset.

- **Simple Co-occurrence Count (SCC)** is just the raw total number of co-occurrences of the chemical concept c with the disease concept d, given by
$$\text{SCC}_{c-d} = N_{c-d} \tag{1}$$
This simple metric does not take into account how often the chemical c and the disease d occur separately.

- **Normalised Point-wise Mutual Information (NPMI)** [24] is the normalised version of the point-wise mutual information that measures the mutual dependence between two variables. In this case the NPMI between the chemical c and the disease d is given by
$$\text{NPMI}_{c-d} = \frac{\log \frac{P(c,d)}{P(c)P(d)}}{-\log P(c,d)} = \frac{\log \frac{N_{c-d}N}{N_c N_d}}{\log \frac{N}{N_{c-d}}} \tag{2}$$
with $\text{NPMI}_{c-d} \in [-1,1]$ where -1 means that c and d never co-occur, 0 means that c and d are independent, and 1 means that c and d always co-occur.

- **Symmetric Conditional Probability (SCP)** [25] is the product of conditional probabilities $P(c|d)$ and $P(d|c)$, where $P(c|d)$ denotes the probability that the chemical c appears in a sentence given that the disease d appeared in that sentence and analogously for $P(d|c)$. Namely,
$$\text{SCP}_{c-d} = P(c|d)P(d|c) = \frac{P(c,d)^2}{P(d)P(c)} = \frac{N_{c-d}^2}{N_d N_c} \tag{3}$$
with $\text{SCP}_{c-d} \in [0,1]$ where 0 means that c and d never co-occur, and 1 means that c and d always co-occur.

- **Dice Coefficient (DC)** [26] is a commonly used association measure for collocation extraction. It represents the intersection of two fuzzy sets [27] and is defined as
$$\text{DC}_{c-d} = \frac{2N_{c-d}}{N_c + N_d} \tag{4}$$
with $\text{DC}_{c-d} \in [0,1]$ where 0 means that c and d never co-occur, and 1 means that c and d always co-occur.

  Comparing to the monotonically related Jaccard coefficient ($\text{JC} = \text{DC}/(2 - \text{DC})$) [28], the DC is not a proper distance metric as it does not satisfy the triangle inequality. This has a benefit that the DC retains sensitivity in more heterogeneous datasets and assigns lower weight to outliers [29]. Furthermore, the DC and its variations such as $\log \text{DC}$ [30] have recently become popular for measuring the lexical association of word pairs. All these findings motivated my decision to use the DC instead of the JC.

  When compared to other association measures, the DC seems to outperform several variations of the mutual information measure in the task of lexicographical collocation extraction, as shown by the works [30, 31]. However, this might be task and data-dependent.

4.3 Results and analysis

In this section I first present the chemical-disease co-occurences extracted using each of the above-defined co-occurence measures, Sec. 4.3.1. I then compare the obtained rankings in terms of their similarity, Sec. 4.3.2. Finally, I analyse the relations of the extracted chemical-disease pairs, Sec. 4.3.3.

4.3.1 Extracted chemical-disease co-occurrences

The top 10 ranked chemical-disease associations for each measure are shown in Tab. 9. Since there might be multiple MeSH entry names corresponding to one MeSH concept, I present the most human-readable variants, for example, *"pneumocystis pneumonia"* instead of *"PCP"*. Among the 4 rankings there are 22 unique chemical-disease pairs (CDPs) with 3 CDPs shared among any 2 rankings, and 6 CDPs shared among any 3 rankings. There is only one CDP that appears in all 4 top 10 rankings, namely, *"levodopa + abnormal movements"*, whose relation I also confirmed by finding this CDP in the list of drug-induced diseases [32].

We can see that the top 10 ranking by SCC considerably differs from the rankings by other measures. For example, the first three chemical-disease pairs (CDPs) by SCC (*"glucose + diabetes"*, and *"cisplatinum + cancer"*, *"doxorubicin + cancer"*) do not even appear in the top 10 ranked by other measures.

- *... In patients with fully developed* Disease NE: *diabetes mellitus* *both, a complete normalisation of* Chemical NE: *glucose* *tolerance as well as no change in the metabolic situation have been observed. ...*
- *... Previous studies have suggested the potentiation of the antitumor activity of certain cytotoxic drugs, such as* Chemical NE: *cisplatin* *and* Chemical NE: *doxorubicin , in human* Disease NE: *cancer* *cell lines ...*

The high numbers N_c and N_d (in the ranking by SCC) further show that concepts c and d occur very often in general and so their co-occurrence in a sentence may not be remarkable. Therefore, this simple metric is unlikely to be useful for discovering *new* chemical-disease associations, as it does not take into account how often the chemical c and the disease d occur separately.

Looking at the ranking by NPMI and the associated counts N_c and N_d we can conclude that it can well pick up the CDPs whose chemical and disease concepts appear rarely on their own and so their co-occurrence is informative. The measure with this property might be useful for discovery of *new* chemical-disease associations.

The top 10 rankings by SCP and DC are very similar and both have the same first two CDPs (”levodopa + abnormal movements” and ”lamivudine + hepatitis B”).

- *... The occurrence of side effects with long-term* Chemical NE: *levodopa* *therapy, such as fluctuations in motor performance or* Disease NE: *abnormal movements , led to a search for new antiparkinsonian drugs. ...*
- *... Recently,* Chemical NE: *lamivudine* *used to treat patients with* Disease NE: *hepatitis B virus (HBV) infection* *was revealed to have potent antiviral activity. ...*

Table 9: Top 10 chemical-disease associations ranked by **SCC**, **NPMI**, **SCP** or **DC** measure (from top to bottom). N_x is the number of sentences where the concept x occurred. *Red:* the only chemical-disease pair present in all 4 top 10 rankings. * denotes the chemical-disease pairs that are annotated as chemically-induced diseases in the *BioCreative V CDR* dataset [4].

Chemical (c) + Disease (d)	N_c	N_d	**SCC**	NPMI	SCP	DC
glucose + diabetes	90297	16278	5062	0.4703	0.0174	0.0950
cisplatinum + cancer	36345	48709	4175	0.4104	0.0098	0.0982
doxorubicin + cancer	38219	48709	4075	0.3997	0.0089	0.0938
alcohol + depression *	250201	19168	3607	0.2597	0.0027	0.0268
levodopa + Parkinson's disease	12578	12467	3371	0.6740	0.0725	0.2692
levodopa + abnormal movements *	12578	9388	3337	0.7071	0.0943	0.3038
carbon tetrachloride + toxic hepatitis *	11471	22211	3148	0.6001	0.0389	0.1869
alcohol + toxic hepatitis *	250201	22211	3061	0.2163	0.0017	0.0225
paracetamol + toxic hepatitis *	17890	22211	2865	0.5276	0.0207	0.1429
estrogen + breast cancer *	22465	17911	2759	0.5191	0.0189	0.1367

Chemical (c) + Disease (d)	N_c	N_d	SCC	**NPMI**	SCP	DC
dapsone + leprosy	1844	290	155	0.7212	0.0449	0.1453
lamivudine + hepatitis B	7543	4002	1587	0.7179	0.0834	0.2749
ribavirin + hepatitis C	4452	1425	626	0.7140	0.0618	0.2130
phencyclidine + pneumocystis pneumonia	2325	3040	665	0.7135	0.0626	0.2479
methylphenidate + attention-deficit/hyperactivity disorder	4166	2407	803	0.7107	0.0643	0.2443
levodopa + abnormal movements *	12578	9388	3337	0.7071	0.0943	0.3038
bicuculline + basal cell carcinoma	2288	82	73	0.7003	0.0284	0.0616
thienodiazepine + massive hepatocellular necrosis	6	23	1	0.6954	0.0072	0.0690
carbon monoxide + poisoning	6854	11122	2398	0.6920	0.0754	0.2668
cinacalcet HCl + hyperparathyroidism	451	551	73	0.6767	0.0214	0.1457

Chemical (c) + Disease (d)	N_c	N_d	SCC	NPMI	**SCP**	DC
levodopa + abnormal movements *	12578	9388	3337	0.7071	0.0943	0.3038
lamivudine + hepatitis B	7543	4002	1587	0.7179	0.0834	0.2749
carbon monoxide + poisoning	6854	11122	2398	0.6920	0.0754	0.2668
levodopa + Parkinson's disease	12578	12467	3371	0.6740	0.0725	0.2692
methylphenidate + attention-deficit/hyperactivity disorder	4166	2407	803	0.7107	0.0643	0.2443
phencyclidine + pneumocystis pneumonia	2325	3040	665	0.7135	0.0626	0.2479
ribavirin + hepatitis C	4452	1425	626	0.7140	0.0618	0.2130
adenosine diphosphate + platelet aggregations	7272	6117	1492	0.6622	0.0500	0.2229
streptozotocin + diabetes *	2712	16278	1484	0.6621	0.0499	0.1563
dapsone + leprosy	1844	290	155	0.7212	0.0449	0.1453

Chemical (c) + Disease (d)	N_c	N_d	SCC	NPMI	SCP	**DC**
levodopa + abnormal movements *	12578	9388	3337	0.7071	0.0943	0.3038
lamivudine + hepatitis B	7543	4002	1587	0.7179	0.0834	0.2749
levodopa + Parkinson's disease	12578	12467	3371	0.6740	0.0725	0.2692
carbon monoxide + poisoning	6854	11122	2398	0.6920	0.0754	0.2668
phencyclidine + pneumocystis pneumonia	2325	3040	665	0.7135	0.0626	0.2479
methylphenidate + attention-deficit/hyperactivity disorder	4166	2407	803	0.7107	0.0643	0.2443
adenosine diphosphate + platelet aggregations	7272	6117	1492	0.6622	0.0500	0.2229
ribavirin + hepatitis C	4452	1425	626	0.7140	0.0618	0.2130
carbon tetrachloride + toxic hepatitis *	11471	22211	3148	0.6001	0.0389	0.1869
isoniazid + tuberculosis	6290	2152	764	0.6703	0.0431	0.1810

4.3.2 Similarity between rankings

To further investigate the similarity between the 4 rankings I calculated the overlap between CDPs for each pair of the *top 10* rankings, as shown in Tab. 10 (left). To examine the similarity between the *full* rankings I also computed the normalised Spearman's Footrule [33] for each pair of rankings obtained by measures x and y. The Spearman's Footrule is defined as the sum of absolute differences in ranks of the ranked items and its normalised form [34] is given by

$$F_{xy} = 1 - \frac{\sum_{i \in \mathbb{S}} |r_x(i) - r_y(i)|}{F_{max}} \qquad F_{max} = \begin{cases} |\mathbb{S}|^2/2 & \text{for } |\mathbb{S}| \text{ even} \\ (|\mathbb{S}|^2 - 1)/2 & \text{for } |\mathbb{S}| \text{ odd} \end{cases} \tag{5}$$

where $\mathbb{S}$ is the set of all ranked CDPs, in my case $|\mathbb{S}| = 46,137$, and r_z maps a CDP to its rank assigned by measure z. The normalised Spearman's Footrule is bounded between 0 (for reversed rankings) and 1 (for identical rankings). The results for all F_{xy} are summarised in Tab. 10 (right). As we can see, the similarities of *full* rankings as well as the similarities of *top 10* rankings confirm the above observations that the SCC measure is most dissimilar to other measures. Next, it is interesting to notice that the most similar rankings according to the *top 10* rankings are generated by SCP and DC whereas the most similar rankings according to the *full* rankings are generated by SCP and NPMI. However, it should be noted that both of the similarity measures presented in Tab. 10 do not take into account the *absolute* positions of a CDP in the two rankings, i.e. whether the CDP appears near the head or near the tail of the ranking lists. Therefore, for further investigation of the similarity between the rankings it might be good to incorporate position weights into the calculation of Spearman's Footrule.

Table 10: Similarity between rankings of chemical-disease pairs (CDPs) made by various co-occurrence measures. *Left:* Number of CDPs shared between two *top 10* rankings. *Right:* Normalised Spearman's Footrule (F_{xy}) for each pair of two *full* rankings made by measures x and y, with $F_{xy} \in [0, 1]$ where 1 means identical rankings.

	SCC	NPMI	SCP	DC
SCC	–	1	2	3
NPMI	1	–	7	6
SCP	2	7	–	8
DC	3	6	8	–

	SCC	NPMI	SCP	DC
SCC	–	0.425	0.609	0.610
NPMI	0.425	–	0.783	0.651
SCP	0.609	0.783	–	0.745
DC	0.610	0.651	0.745	–

4.3.3 Chemical-disease relations

The chemical-disease associations might arise for several reasons such as a chemical causes a disease or a disease treated by a chemical. To investigate the former CAUSES relation, I evaluated how many and which of the top 10 CDPs represent a chemically-induced disease (CID). For this I used the *BioCreative V CDR* dataset [4] (introduced in Sec. 2) that provides annotated CIDs. In total, I extracted 3,116 such CIDs from train, development and test partitions and took the set intersection with the top 10 CDPs ranked by various measures. This overlap is summarised in Tab. 11 and the CDPs annotated as CIDs are also highlighted in Tab. 9 by *. We can conclude that, regarding the number of CIDs present in the top 10 ranking, the SCC measure performs the best. This suggests that SCC might be useful for identification of chemically-induced diseases.

Table 11: Intersection of the top 10 ranked chemical-disease pairs (CDPs) and the annotated chemically-induced diseases (CIDs) from the *BioCreative V CDR* dataset [4]. *Left:* CIDs appearing in top 10 CDPs aggregated from all 4 rankings. *Right:* number of CIDs appearing in top 10 CDPs ranked by various measures.

Chemical + Disease
paracetamol + toxic hepatitis
alcohol + depression
alcohol + toxic hepatitis
carbon tetrachloride + toxic hepatitis
estrogen + breast cancer
levodopa + abnormal movements
streptozotocin + diabetes

Co-occurrence measure	Number of intersecting chemically-induced diseases
SCC	**6**
NPMI	1
SCP	2
DC	2

To examine the CDPs in terms of other possible relations between chemicals and diseases, I consulted the list of 22 unique CDPs (aggregated from all 4 top 10 rankings) with the physician MUDr Maria Kleinova. In particular, she classified the list of unique CDPs according to the relation between the chemical c and disease d, defining the following set of relations

$$c \text{ CAUSES } d$$
$$c \text{ TREATS } d$$
$$c \text{ DETECTS } d$$
$$c \text{ INCREASES RISK OF } d$$
$$c \text{ ACCOMPANIES } d$$
$$c \text{ UNKNOWN } d$$

where UNKNOWN means that even though she knew the individual chemicals and diseases she was not aware of any relations between them. Following her suggestions I allowed multiple relations to be assigned to a given CDP. The categorisation of the unique CDPs according to these relations is presented in Tab. 12. We can see that most of the CDPs exhibit the TREATS or CAUSES relation and that the relation of two CDPs is UNKNOWN. It seems to be confusing that *"levodopa + abnormal movements"* is assigned with two opposite relations. However, as explained by MUDr Kleinova, the reason is that *levodopa* is usually used to treat *abnormal movements* but on the other side it can also have negative side-effects of *abnormal movements*. This is also confirmed by the sample extracted sentence shown in Sec. 4.3.1 and agrees with the classification of *"levodopa + Parkinson's disease"* to have only the TREATS relation. As a sanity check, we can ensure that the CDPs previously identified as CIDs in Tab. 11 are assigned with relation types that are in agreement with a more general INDUCES relation of CIDs.

Table 12: Chemical-disease relations of the unique chemical-disease pairs (CDPs) aggregated from all 4 top 10 rankings, as classified by the physician MUDr Maria Kleinova.

Chemical + Disease	Relation type
levodopa + abnormal movements	CAUSES / TREATS
carbon monoxide + poisoning	CAUSES
ribavirin + hepatitis C	TREATS
lamivudine + hepatitis B	TREATS
levodopa + Parkinson's disease	TREATS
methylphenidate + attention-deficit/hyperactivity disorder	TREATS
phencyclidine + pneumocystis pneumonia	UNKNOWN
dapsone + leprosy	TREATS
adenosine diphosphate + platelet aggregations	DETECTS
carbon tetrachloride + toxic hepatitis	CAUSES
estrogen + breast cancer	INCREASES RISK OF
cisplatinum + cancer	TREATS
isoniazid + tuberculosis	TREATS
paracetamol + toxic hepatitis	CAUSES
cinacalcet HCl + hyperparathyroidism	TREATS
bicuculline + basal cell carcinoma	UNKNOWN
streptozotocin + diabetes	CAUSES
doxorubicin + cancer	TREATS
alcohol + toxic hepatitis	CAUSES
thienodiazepine + massive hepatocellular necrosis	CAUSES
glucose + diabetes	ACCOMPANIES
alcohol + depression	CAUSES / ACCOMPANIES

I further analysed what types of relations the top 10 CDPs have when ranked by various co-occurrence measures. The results in Tab. 13 indicate that SCC is best suited for extraction of CDPs with CAUSES relation whereas the other three measures seem to be better at mining CDPs with the TREATS relation. As we can see, the NPMI measure performs best at finding CDPs with UNKNOWN relations, which agrees with the suggestion from Sec. 4.3.1 that the NPMI measure is likely to be useful for discovery of *new* chemical-disease associations. However, such conclusions should be taken with care and further investigation should be performed, as I have found some sentences where the chemical-disease co-occurrence was incorrectly identified because of the ambiguity between the abbreviation of the chemical and the abbreviation of the disease concept. For example, in the below sentence the abbreviation for the *"basal cell carcinoma"* disease is confused with the abbreviation of the chemical *"bicuculline"* and so the CDP *"bicuculline + basal cell carcinoma"* is incorrectly identified.

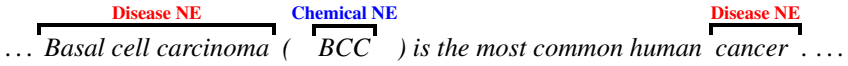


Table 13: Number of top 10 ranked chemical-disease pairs (CDPs) with particular type of chemical-disease relation, when ranked by various co-occurrence measures. Sum of counts per row may exceed 10 since some CDPs were assigned multiple relation types, see Tab. 12.

	Relation type					
Measure	CAUSES	TREATS	DETECTS	INCREASES RISK OF	ACCOMPANIES	UNKNOWN
SCC	**5**	4	0	1	2	0
NPMI	3	6	0	0	0	**2**
SCP	3	6	1	0	0	1
DC	3	6	1	0	0	1

5 Conclusion

In this project I analysed the co-occurrences of chemicals and diseases in the biomedical literature using various co-occurrence measures. I trained and tuned the named entity recognition model based on Conditional Random Fields to identify chemical and disease named entities (NEs) in text, and designed a grounding system to normalise the recognised NEs to Medical Subject Heading (MeSH) concepts.

My results show that the chemical-disease pair (CDP) *"levodopa + abnormal movements"* appears in top 10 rankings by each of the examined co-occurrence measures. My investigation further shows that the Simple Co-occurrence Count (SCC) is unlikely to be useful for discovering *new* chemical-disease associations whereas the Normalised Point-wise Mutual Information (NPMI) is promising for this task. Also, the ranking of CDPs by SCC measure is most dissimilar to rankings by other measures. Regarding the type of the relation between chemicals and diseases, the SCC measure seems to be best suited for identification of chemically-*induced* diseases, while the NPMI measure for extraction of CDPs with *not very well known* or possibly *unknown* relations.

The possible directions for further improvement of the NER model and the SASM-based grounding technique were already suggested in Sec. 2.5 and at the end of Sec. 3.1 respectively.

References

[1] Rezarta Islamaj Dogan, G. Craig Murray, Aurélie Névéol, and Zhiyong Lu. Understanding pubmed® user search behavior through log analysis. *Database*, 2009:bap018, 2009.

[2] Allan Peter Davis, Cynthia J Grondin, Robin J Johnson, Daniela Sciaky, Benjamin L King, Roy McMorran, Jolene Wiegers, Thomas C Wiegers, and Carolyn J Mattingly. The comparative toxicogenomics database: update 2017. *Nucleic acids research*, 45(D1):D972–D978, 2016.

[3] Jinghang Gu, Longhua Qian, and Guodong Zhou. Chemical-induced disease relation extraction with various linguistic features. *Database*, 2016, 2016.

[4] Chih-Hsuan Wei, Yifan Peng, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Jiao Li, Thomas C Wiegers, and Zhiyong Lu. Overview of the biocreative v chemical disease relation (cdr) task. In *Proceedings of the fifth BioCreative challenge evaluation workshop*, pages 154–166. Sevilla Spain, 2015.

[5] Jun Xu, Yonghui Wu, Yaoyun Zhang, Jingqi Wang, Hee-Jin Lee, and Hua Xu. Cd-rest: a system for extracting chemical-induced disease relation in literature. *Database*, 2016, 2016.

[6] George R Krupka and Kevin Hausman. Isoquest inc.: Description of the netowl (tm) extractor system as used for muc-7. In *Proceedings of MUC*, volume 7, 1998.

[7] Satoshi Sekine and Chikashi Nobata. Definition, dictionaries and tagger for extended named entity hierarchy. In *LREC*, pages 1977–1980. Lisbon, Portugal, 2004.

[8] Radu Florian, Abe Ittycheriah, Hongyan Jing, and Tong Zhang. Named entity recognition through classifier combination. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 168–171. Association for Computational Linguistics, 2003.

[9] Sameer Singh, Dustin Hillard, and Chris Leggetter. Minimally-supervised extraction of entities from text advertisements. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 73–81. Association for Computational Linguistics, 2010.

[10] Rohini Srihari, Cheng Niu, and Wei Li. A hybrid approach for named entity and sub-type tagging. In *Proceedings of the sixth conference on Applied natural language processing*, pages 247–254. Association for Computational Linguistics, 2000.

[11] Martin Jansche and Steven P Abney. Information extraction from voicemail transcripts. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 320–327. Association for Computational Linguistics, 2002.

[12] John Lafferty, Andrew McCallum, and Fernando CN Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. 2001.

[13] Peilu Wang, Yao Qian, Frank K Soong, Lei He, and Hai Zhao. A unified tagging solution: Bidirectional lstm recurrent neural network with word embedding. *arXiv preprint arXiv:1511.00215*, 2015.

[14] Zhiheng Huang, Wei Xu, and Kai Yu. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*, 2015.

[15] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*, 2016.

[16] Xuezhe Ma and Eduard Hovy. End-to-end sequence labeling via bi-directional lstm-cnns-crf. *arXiv preprint arXiv:1603.01354*, 2016.

[17] Thierry Poibeau and Leila Kosseim. Proper name extraction from non-journalistic texts. *Language and computers*, 37:144–157, 2001.

[18] Naoaki Okazaki. Crfsuite: a fast implementation of conditional random fields (crfs), 2007.

[19] Dong C Liu and Jorge Nocedal. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45(1-3):503–528, 1989.

[20] Andrew McCallum and Wei Li. Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 188–191. Association for Computational Linguistics, 2003.

[21] Md Shad Akhtar, Utpal Kumar Sikdar, and Asif Ekbal. Iitp: Hybrid approach for text normalization in twitter. In *Proceedings of the Workshop on Noisy User-generated Text*, pages 106–110, 2015.

[22] Vladimir I Levenshtein. Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*, volume 10, pages 707–710, 1966.

[23] Predictivities Based ROC Curve. Medicalbiostatistics. com.

[24] Gerlof Bouma. Normalized (pointwise) mutual information in collocation extraction. *Proceedings of GSCL*, pages 31–40, 2009.

[25] Joaquim Ferreira Da Silva, Gaël Dias, Sylvie Guilloré, and José Gabriel Pereira Lopes. Using localmaxs algorithm for the extraction of contiguous and non-contiguous multiword lexical units. In *Portuguese Conference on Artificial Intelligence*, pages 113–132. Springer, 1999.

[26] Lee R Dice. Measures of the amount of ecologic association between species. *Ecology*, 26(3):297–302, 1945.

[27] David W Roberts. Ordination on the basis of fuzzy set theory. *Vegetatio*, 66(3):123–131, 1986.

[28] Paul Jaccard. The distribution of the flora in the alpine zone. *New phytologist*, 11(2):37–50, 1912.

[29] Bruce McCune, James B Grace, and Dean L Urban. *Analysis of ecological communities*, volume 28. MjM software design Gleneden Beach, OR, 2002.

[30] Pavel Rychlỳ. A lexicographer-friendly association score. *Proceedings of Recent Advances in Slavonic Natural Language Processing, RASLAN*, 2008:6, 2008.

[31] Dipak L Chaudhari, Om P Damani, and Srivatsan Laxman. Lexical co-occurrence, statistical significance, and word association. In *Proceedings of the conference on empirical methods in natural language processing*, pages 1058–1068. Association for Computational Linguistics, 2011.

[32] Vishal R Tandon, Vijay Khajuria, Vivek Mahajan, Aman Sharma, Zahid Gillani, and Annil Mahajan. Drug-induced diseases (dids): An experience of a tertiary care teaching hospital from india. *The Indian journal of medical research*, 142(1):33, 2015.

[33] Persi Diaconis and Ronald L Graham. Spearman's footrule as a measure of disarray. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 262–268, 1977.

[34] Susanne Mikki. Comparing google scholar and isi web of science for earth sciences. *Scientometrics*, 82(2):321–331, 2010.

A Feature ablation experiments

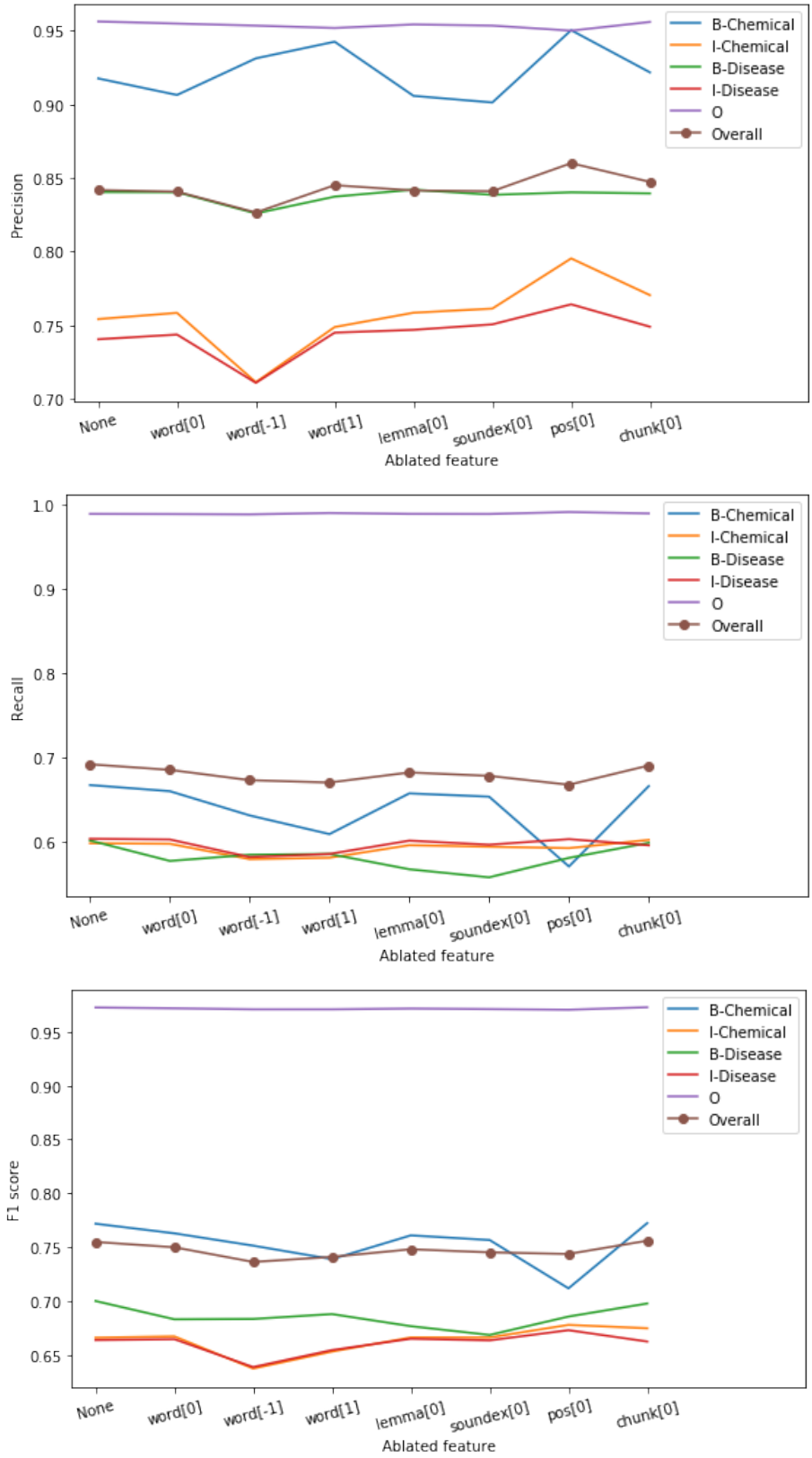


Figure 6: Feature ablation experiments: **precision**, **recall** and **F1 score** (from top to bottom) for each item class and overall (macro-average). Evaluated on development set.