Unsupervised Method for Disease Named Entity Recognition

Thesis by
Abeer Nayer Almutairi

In Partial Fulfillment of the Requirements
For the Degree of
Masters of Science

King Abdullah University of Science and Technology
Thuwal, Kingdom of Saudi Arabia

November, 2019
The thesis of Abeer Nayer Almutairi is approved by the examination committee

Committee Chairperson: Robert Hoehndorf
Committee Members: Mikhail Moshkov, Taous-Meriem Laleg-Kirati
ABSTRACT

Unsupervised Method for Disease Named Entity Recognition
Abeer Nayer Almutairi

Diseases take a central role in biomedical research; many studies aim to enable access to disease information, by designing named entity recognition models to make use of the available information. Disease recognition is a problem that has been tackled by various approaches of which the most famous are the lexical and supervised approaches. However, the aforementioned approaches have many drawbacks as their performance is affected by the amount of human-annotated data set available. Moreover, lexical approaches cannot distinguish between real mentions of diseases and mentions of other entities that share the same name or acronym. The challenge of this project is to find a model that can combine the strengths of the lexical approaches and supervised approaches, to design a named entity recognizer. We demonstrate that our model can accurately identify disease name mentions in text, by using word embedding to capture context information of each mention, which enables the model to distinguish if it is a real disease mention or not. We evaluate our model using a gold standard data set which showed high precision of 84% and accuracy of 96%. Finally, we compare the performance of our model to different statistical name entity recognition models, and the results show that our model outperforms the unsupervised lexical approaches.
ACKNOWLEDGEMENTS

I would like to express my sincere appreciation to my family for their emotional support and encouragement, especially my husband who has supported me in so many different ways. I then extend my appreciation to my best friend Sumyyah for her spiritual and academic support. I am also thankful to my friends: Sara, Mona, Sarah, and Azza because they always enrich my knowledge with their discussions and suggestions.

I dedicate this work to my parents, I wish I always make you happy and proud, I’m grateful for your prayers and encouragements. I also dedicate this work to my husband for his generosity and limitless love.

I hold a great amount of gratitude towards my advisor Professor Robert Hoehndorf for his guidance, support, and encouragement throughout this journey. I am thankful for his constant follow up, feedback and suggestions.
# TABLE OF CONTENTS

Examination Committee Page 2

Copyright 3

Abstract 4

Acknowledgements 5

List of Figures 8

List of Tables 9

1 Introduction 10

2 Literature Review 13
   2.1 Lexical approaches ........................................... 13
   2.2 Supervised approaches ........................................ 15
   2.3 Ontologies ...................................................... 17
   2.4 Word embedding ............................................... 18
       2.4.1 Word2vec ................................................ 18
       2.4.2 BERT ..................................................... 19

3 Problem Statement 20

4 Materials and Methods 22
   4.1 Data resources ................................................ 22
       4.1.1 Corpus .................................................... 22
       4.1.2 Disease resources ...................................... 22
   4.2 Lexical baseline .............................................. 23
   4.3 BERT ......................................................... 24
   4.4 K-mean clustering ............................................ 25
   4.5 Supervised training ......................................... 26
5 Results & Discussion

5.1 Testing process ................................................. 30
5.2 Evaluation ....................................................... 31

6 Concluding Remarks

References ............................................................ 37

Appendices ............................................................ 42
LIST OF FIGURES

4.1 The lexical baseline architecture .................................................. 23
4.2 BERT ..................................................................................... 24
4.3 Final model .............................................................................. 27
5.1 K-mean clustering .................................................................. 30
5.2 Testing procedure .................................................................. 31
LIST OF TABLES

4.1 Size of the NCBI disease corpus ........................................ 22
5.1 K-Mean cluster with K=2 ............................................ 29
5.2 K-Mean cluster with K=3 ............................................ 29
5.3 K-Mean cluster with K=4 ............................................ 29
5.4 K-Mean cluster with K=5 ............................................ 29
5.5 Performance of our model ............................................ 32
5.6 Performance compared against lexical baselines ................. 34
5.7 Performance compared against another supervised model .... 34
Chapter 1

Introduction

There is an ever-increasing amount of biomedical literature in the field of biomedical research. Therefore, it is very important to focus on biomedical text mining, to make use of the available information [1]. Biomedical text mining is defined by the extraction and discovery of high-quality, non-trivial information from free text [2].

One of the main components of biomedical text mining is Biomedical Natural Language Processing (BioNLP), which allows computers to extract information about biomedical topics from unstructured texts, such as the type of entities which are mentioned in texts [1].

In BioNLP, the name of biological entities mentioned in texts need to be automatically identified, which is known as Named Entity Recognition (NER) [3]. NER identifies mentions of named entities, which are predefined meaningful names in particular domains, as well as the particular domain of the entities from texts [1].

The main focus of this study, is to design a name entity recognizer that can identify mentions of disease names in texts. Mentions of name entities are terms of disease names but within their context. A term could be a word defined by a single distinct element of speech or writing. Also, a term could be a phrase, which is multiple words without the subject-verb pair necessary to form a clause [5]. Given below is an example to illustrate the idea:

- Terms could be a disease name consisting of one or multiple words, such as:
  - cancer
- breast cancer

- Mentions are terms of disease names but within their context, such as:

  - If your mother had breast cancer, the chances of getting the disease is higher.

Diseases take a central role in biomedical research; many studies aim to enable access to disease information [6], since many biomedical applications need to discover the available information about diseases for the sake of disease detection, diagnosis, prevention, and treatment. Other applications depend on the information extracted such as drug side effect detection, safety surveillance and drug discovery [7].

Diseases are defined by any impairment of normal biological function [8]. Disease names are terms created by combining disease category with modifiers, which take many forms. They are usually used as a short descriptive modifier, including symptoms (dry-eye syndrome), anatomical locations (ovarian cancer), causative agent (oral thrush infection), treatment (dopamine responsive dystonia), heredity (Bruton agammaglobulinemia), or eponyms (Alzheimer’s disease). Furthermore, modifiers provide descriptions such as severe malaria, which are not part of the name [9].

Disease names are created from Latin or Greek names by combining affixes and roots such as hemochromatosis. When disease names are mentioned in the text, they exhibit variations, and this variation is presented in the abbreviation, synonymous terms for the same disease, and the usage of different word orderings. These variations may be morphological or orthographical, and they involve more than one-word substitutions. For example, a disease name could be composed by combining affixes such as a single word (cancer) that may correspond to multiple words (breast, brain and ovarian) in another form [9].

Disease names include labels, synonyms, and abbreviations, and they can be used to refer to a unique disease class, and every class is identified by a unique class ID.
Consequently, disease name could be mentioned in two different places or sentences. Then, we consider each of these two different places or sentences as a different mention for the same disease name. It is the unique combination of a term and its context, which form a mention. For instance, in the following two sentences:

- This is breast cancer disease
- Breast cancer patients are in danger

Breast cancer is the disease name term, and each of these sentences is considered as a different mention.

Here, we develop a novel method that combines the strengths of lexical approaches and supervised approaches to design a name entity recognizer. We demonstrate that our method can be used for any texts. Specifically, the contributions in this thesis are as follows:

- We develop a novel model for disease name entity recognition.
- We identify disease names mentioned in text.
- We evaluate our findings against different statistical name entity recognizer models.
Chapter 2

Literature Review

2.1 Lexical approaches

Traditional biomedical name entity recognizer strategies mainly depend on rule-based systems or dictionary lookups. Rule-based systems define rules manually using textual patterns to recognize biomedical entities. A rule-based system was introduced by Narayanaswamy et al. [10], and it was manually developed using a set of rules that depend on lexical information and English linguistic constraints. Named entities were categorized using contextual information and surrounding words. Furthermore, Budi & Bressan [11] proposed a new rule-based system to extract named entities using a set of grammatical and syntactic rules. Rule-based systems depend on two types of rules [12]: pattern-based rules, which depend on the word patterns being orthographic or morphological, and context-based rules, which depend on the contextual information in the text.

Rule-based systems mainly depend on the specific textual properties, and it is time-consuming to define such entities with specific textual properties and patterns. Also, it requires experts with specific domain knowledge. When the resources are available the rule-based system performs well, but the system will lack portability if the data are slightly changed, since it is highly costly to maintain the status of the rule updated.

A dictionary is built manually or automatically by collecting vocabulary for a specific domain from related databases. There are different examples of dictionaries,
such as the Human Disease Ontology (DO) [13], which is used to identify disease names in text, and the DrugBank [14] dictionary, which is used to identify names of drugs. These dictionaries are used to implement string-matching algorithms, and they identify entity occurrences in the text using lists of terms defined in dictionaries. This process makes an exact search if a term in the text matches a term in the dictionary.

Dictionary-based systems are used for named entity extraction by searching for terms in dictionaries, and looking for dictionary term matches in texts. For instance, Hettne et al. [15] used a dictionary-based approach to identify name entities of small molecules and drugs in texts, combining multiple dictionaries to increase the power of recognizing chemical name entities and map them to thier class ID. Gerner et al. [16] introduced LINNAEUS, which it is a dictionary-based software system used to recognize species names.

Two factors that greatly impact the quality of the dictionary-based systems, are the quality of both the dictionary and the matching algorithm used. Furthermore, it is time-consuming to build and maintain up-to-date dictionaries. Generally, dictionary-based systems offer poor recall, since it is difficult to capture new entities and to avoid spelling errors in the text. On the other-hand, dictionary-based methods can directly map the mention to their class ID.

Lexical approaches affect precision since it they keep the longest string in case of nested term mentions. Also, many terms may be ambiguous or abbreviated. Furthermore, lexical approaches are expected to have difficulty dealing with term variability. Many resources [17, 18, 19] are designed to address issues concerning case normalization, word order, and plurals.

The hybrid system combines dictionary-based systems and rule-based systems to design a named entity recognizer. For example, the LeadMine system [20] employs spelling correction and merges the adjacent entities and entity extension to improve performance. This increases the chance of recognizing trivial names that are slightly
different from the list of terms covered in the dictionary.

2.2 Supervised approaches

In recent years, lexical approaches like rule-based or dictionary-based methods have been replaced by supervised approaches. The basic steps required for systems are training the model to use the present annotations from the annotated documents and to annotate the documents to generate the entity names based on past knowledge learned from the annotated documents.

Supervised learning algorithms learn from the labeling of the training data-set with the correct outcomes. For instance, the computer learns from the input of the classification problems to create a classification system that produces output accordingly [21].

Supervised model quality depends on the available inputs in the training data set, since the construction process to build the supervised model is affected by the amount of missing inputs. There are two processes to construct the supervised model. The first process is the feature selection and extraction, and many environments can be used to transform and extract features, such as the frameworks described by [22, 23, 24]. The second process is selecting the subset of features to represent a certain label.

In the feature selection process, feature redundancy is not required since it does not provide any more information. Furthermore, using irrelevant features does not provide useful information. Usually, to improve the primary features set, a set of experiments is carried out by deleting, adding or modifying features [22]. Also, the combinations of features could increase the performance of the model [25].

Lafferty et al [26] designed Conditional Random Fields (CRFs) for sequential classification. CRFs are undirected statistical graphical models conditionally trained to segment and label sequence data. The named entity recognition problem can be
described as a sequence classification problem. Thus, the CRFs framework could be used to assign labels to words in a sentence.

A Biomedical Named Entity Recognizer (ABNER) [27] is used for molecular biology text mining. It is an open-source software tool with a graphical user interface, and each user can use the tool by typing texts manually or automatically to highlight the biological name entities. This system was built using linear-chain CRFs with other features such as contextual and orthographic, which mostly depend on term context or regular expressions.

BANNER [28] is a supervised machine-learning system use CRFs for labeling and feature generation, such as context with a window of two and part-of-speech tagging as features. BANNER could be extended to build another named entity recognizer, such as DNorm [9]. DNorm uses BANNER to locate disease name mentions, and the features are based on a dictionary derived from the Unified Medical Language System (UMLS) [29], and then the output mentions are used for additional string processing. As an example, the short-form abbreviations found in mentions are replaced with their long form. Also, punctuation and stop words are removed, and tokens are then stemmed using Porter Stemmer [30]. In the final step to generate a candidate for each mention, all the mentions and the disease names in MEDIC are represented in the vector space, and then they search for the maximum scoring function from the training data.

ChemSpot [31] is a novel machine-learning framework based on CRFs and a dictionary. It is used as a named entity recognizer to identify chemical mentions in natural language texts. Employing hybrid NER approaches by combining CRFs and a dictionary is highly beneficial. CRFs are used to extract morphologically chemical name entities, whereas a dictionary is used to extract trivial names, and short and structured names of drugs.

All the previous methods need predefined sets of features, and the extraction
of features from the gold standard annotations. These methods highly depend on training data to recognize name entities. However, the construction of gold standard corpora is a difficult, time-consuming process. Also, the size and quality of the annotations are highly affected by the availability of task-specific annotators which directly influence the development of the gold standard corpora [32].

Some studies, such as [33, 34] have examined the impact of combining features using domain-independent feature sets, such as contextual and lexical features. Using a combination of different feature sets effectively improves the performance of the designed NER system. Furthermore, considering orthographic and morphological features, such as capitalization, punctuation, word shape patterns and stop words, is very important in designing supervised model approaches, since it yields promising results. For example, combining different sets of features is advantageous, since they detect the named entity boundaries [34].

The CheNER-BioC [35] is a chemical named entity recognizer. It uses a hybrid system that employs CRFs together with regular expression taggers and dictionaries, to identify chemical entities in the biomedical literature. The system performs better than when only one approach is used in the designing process.

2.3 Ontologies

Ontologies could be defined as a domain vocabulary, where each ontology class and relation is associated with a list of terms. In addition, ontologies could provide descriptive information such as textual descriptions and definitions to describe what type of things refer to a class or relation [36]. In this study, the Human Disease Ontology (DO) [37] was used to identify what kind of class each disease name refers to. It is a comprehensive vocabulary domain for human disease representation. The following example illustrates a class from Human Disease Ontology (DO): [37]:

- The class id is DOID_1781.
• The class label is thyroid cancer.

• The class synonyms are a malignant tumor of thyroid gland|thyroid neoplasm|malignant neoplasm of thyroid gland|neoplasm of thyroid gland|thyroid gland neoplasm.

2.4 Word embedding

Word embeddings are mathematical objects that represent the words using an unsupervised approach, often called word vectors. They are very good at capturing information about word co-occurrences and therefore their semantic and syntactic characteristics in language. This characterizes the relationship between words by a specific relation vector \[38\]. Several NER tasks \[39, 40\] successfully use word embeddings as features for training purposes. Some examples of word embedding models are listed below.

2.4.1 Word2vec

*Mikolov et al.* \[41\] designed the Word2Vec model, which is a word embedding model, where each word is represented by a vector. This is useful in grouping the vectors of similar words in the same vector space. Moreover, word2Vec provides two architectures: Continuous Bag-Of-Words (CBOW) and Skip-Gram (SG). The Word2Vec model does not consider the order of the word in the context.

The CBOW architecture uses neighboring words to predict each word vector. It uses a bag of words to represent the input layer. The order of the words in the context does not affect the projection of the word vector in the vector space. The SG architecture uses a single word to predict the context of the neighboring words, and it can be used in sentences to predict the neighboring words. Theoretically, SG usually gives higher accuracy for rare or infrequent words. On the other hand, CBOW works well with a large amount of data and gives better accuracy for frequent words.
2.4.2 BERT

Bidirectional Encoder Representations from Transformers (BERT) [42] is a new word embedding model. It is designed to jointly condition left and right context to pre-train deep bidirectional representations from the unlabeled text in all layers. As a result, the BERT model can be used to create models for a variety of tasks, such as named entity recognition and question answering problems. The objective of the model is to predict word embeddings based on the context, and it uses the left and right context and the order of the word to generate word vector.
Chapter 3

Problem Statement

There are many advantages and disadvantages associated with lexical approaches. On the positive side, lexical approaches use dictionaries which are hand-constructed, and they capture rules and regulations that are not discovered statistically [43]. Furthermore, lexical approaches might use rule-based systems structured by a defined set of rules. On the other hand, lexical approach systems depend highly on the pre-defined rules or names, and this could affect the system quality because many disease names are lost or not recognized since they are not supported by the system or they might not exist in the dictionary used. Also, lexical approaches do not use context to distinguish if the mention found is a real mention or not.

One successful approach to address the named entity recognition is to use a supervised learning approach, which requires good preparation of the training data to classify the set of labels. However, using this approach has two major drawbacks: First, it requires a human-annotated data set. Second, the performance is highly dependent on the amount of the labeled training data set available [44].

The challenge of this project is to find a method that can use the strengths of lexical approaches and supervised approaches, to design a named entity recognizer.

Our first hypothesis is that disease names appear in a certain context structure. As a result, the context of mention can be used to determine whether a term refers to a disease name or not, for example:

- Cancer is a group of diseases involving abnormal cell growth.
- Cancer is an astrological sign.

From the given examples, the term cancer term was used to refer to a disease name only in the first example based on the context, while the term "cancer" was not used as a disease name in the second example.

Our second hypothesis is that many disease names in the disease ontology are very specific, clear and unique and cannot be used to refer to any other entity than to the disease itself. As a result, if most of the disease names are used, they will only refer to a disease, for example, retinoblastoma, ataxia-telangiectasia, and anemia. In contrast, a few disease names in the Disease Ontology occur very frequently and when they are used they usually do not refer to a disease, for example cancer, ADD, and trauma.
Materials and Methods

4.1 Data resources

4.1.1 Corpus

The National Center for Biotechnology Information (NCBI) disease corpus [45], is used in the task of disease named entity recognition as a gold standard resource. The research resource is fully annotated in the mentioned level for 693 PubMed abstracts, where each abstract was manually annotated for disease mentions by two human annotators. For the annotation purposes each disease mention was annotated using MeSH [46] and OMIM [47]. Pre-annotations were done as a pre-step using PubTator [48] before the manual annotation step. The corpus is divided into two subsets as described in Table 4.1.

<table>
<thead>
<tr>
<th>Setup</th>
<th>Abstracts</th>
<th>Mentions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training subset</td>
<td>593</td>
<td>5145</td>
</tr>
<tr>
<td>Test</td>
<td>100</td>
<td>960</td>
</tr>
</tbody>
</table>

4.1.2 Disease resources

The Human Disease Ontology (DO) [37] was used to identify disease name mentions. In addition to this, MEDIC was used as a disease vocabulary resource, because it merges Online Mendelian Inheritance in Man (OMIM) [47] with the diseased part of Medical Subject Headings (MeSH) [46]. It was created to index diseases in biomedical
literature and to find relations between the disease names and their classes by the Comparative Toxicogenomic Database.

4.2 Lexical baseline

The first task of the project was to automatically identify the disease name mentions in texts. My hypothesis was to build a lexical baseline to see how it would perform on the NCBI training disease corpus [45]. The lexical baseline architecture is illustrated in Figure 4.1 where the input for the was my NCBI training disease corpus where I created my dictionary using the DO and the Comparative Toxicogenomic Database because it combines MESH and OMIM.

![Figure 4.1: The lexical baseline architecture](image)

This baseline was designed to find the full exact string matches between the texts in question and the mappable dictionaries. All texts were normalized by, lowercasing all the letters. Also, we removed the noise by removing all the stop words. However, before this step, we ignored the stop words that exist in the disease ontologies, such as (IS) and (AND).

The output from the lexical baseline was a list of mentions representing the matches between the text and the dictionary. After some investigations on the extracted list of mentions, it was clear that lexical baselines cannot distinguish between
disease mentions and non-disease mentions using context. An example will be given to illustrate:

- The amount of DNA required to be searched for novel coding sequences constitutes the **HFE** defect.
- Markers display no significant association with **HFE**.

The lexical baseline annotates 'HFE' in both mentions while it is only used as a disease in the first mention.

### 4.3 BERT

To further distinguish between diseases and entities that are very similar to each other but are non-diseases, based on our first hypothesis, it was found that disease names appear in certain context structures. The context of the mention was used to determine whether a term refers to a disease name or not, by utilizing the BERT model [42], because it is proven that BERT could capture the different context in which a word may appear.
The BERT model was used to identify the different contexts in which the disease names appear. Figure 4.2 shows that the input to the BERT model is the NCBI training disease corpus, where the pre-trained model of BERT on the Pubmed corpora was used, to generate BERT word embeddings by using the context of each abstract in the NCBI training corpus [45]. The output is the list of vectors representing the list of mentions extracted from the lexical baseline.

4.4 K-mean clustering

The K-mean cluster divides a given data set into a certain number of clusters. The main idea of the algorithm is to define k centers, one for each cluster [49]. In order to understand the hidden patterns and structures of BERT vectors, the K-mean clustering algorithm was used on the BERT [42] vectors of the disease name mentions, while varying the K with different numbers 2, 3, 4 and 5.

The results of the generated clusters by the algorithm was documented, and each of the clusters contained a different number of disease mentions according to the number of clusters.

According to our second hypothesis, many disease names in the disease ontology are very specific and unique and cannot be used to refer to any other entity than to the disease itself. Also, they are not ambiguous. As an example, it is rare to find a specific disease name like ‘urogenital abnormalities’ when it is not used as a disease name. On the other hand, a few diseases have ambiguous terms and they are frequently mentioned in different contexts. For instance, acronyms, abbreviations, and some words can be used to refer to diseases and other entities as well.

We have done some manual analysis of the clusters, to find how many unique disease names are mentioned in each cluster. In our hypothesis, the cluster which contains the few diseases that occur frequently, does not refer to a disease when the disease mentions are used. On the other hand, if we had a large amount of unique
disease names in the cluster, we consider this cluster as a true cluster, that contains clear and specific disease names, because it is rare to find specific disease names used frequently with a different context.

Based on the above-mentioned hypothesis, and to reflect our second hypothesis, we used a ratio, as a measurement to see how many unique terms exist in each cluster. The equation for the ratio is:

$$\text{Ratio} = \frac{\text{Number of unique terms}}{\text{Number of mentions}}$$  \hspace{1cm} (4.1)

After calculating the ratio for each cluster, we could say that a low ratio means that the cluster contains generic words that are frequently mentioned in a different context. On the other hand, a high ratio means the cluster contains real disease mentions, where only disease names are used.

### 4.5 Supervised training

The final model architecture is shown in Figure 4.3, where a supervised model was trained to predict whether a disease name mention refers to a disease or not.

We used the cluster with the lowest ratio to represent the negatives, and the cluster with the highest ratio to represent the positives, to train our logistic regression [50]. As a result, all the disease mentions in the cluster with the highest ratio are positives, while all the disease mentions in the cluster with the lowest ratio were used as negatives, which refer to non-diseases.

Our final model used the supervised classifier, with an additional filter to finally distinguish between disease mentions and non-disease mentions. The additional filter is dictionary-based, and was created using DO and the Comparative Toxicogenomic Database. If the filter found an exact match in the dictionary, the matches were immediately classified as positives. Moreover, the filter made approximate matching
for the ones classified as positives from the classifier, but not exactly matched in the dictionary, by using the edit distance and the difference in the last character as a measurement of matching. As an example, if 'cancers' is classified as a disease in my classifier but does not exactly match any term in the dictionary, we calculate the edit distance and we check if the difference is on the last character. Then, we consider it a positive match.
Chapter 5

Results & Discussion

The lexical baseline was used to extract disease mentions from the NCBI training corpus [45]. The baseline uses the full exact string-matching method to find matches between the texts in question and the mappable dictionary, that was built using disease ontologies. 3028 disease mentions were extracted from the NCBI training corpus [45].

The aim of the next step was to distinguish the context between diseases and entities that are very similar to them but are false. BERT [42] word embedding was used, to generate vector representation of each disease name mention found in the abstract by the lexical baseline. Since 3028 disease mentions were extracted from the NCBI training corpus [45], 3028 vector representations of each disease name mention were generated based on the context.

To see if the vectors are distinguishable, K-mean cluster was used while varying the K with different numbers 2, 3, 4 and 5. Each of the 3028 vector representations of the disease name mentions extracted by the lexical baseline and represented by BERT, were clustered into different numbers. The results of the clusters generated by the algorithm are shown in Tables 5.1, 5.2, 5.3, 5.4. A manual analysis of the clusters was done, to find how many unique disease names are mentioned in each cluster, and to find the ratio between the number of unique diseases mentioned in a cluster and the total number of disease names in the cluster.

As a result, a low ratio means that the cluster contains generic words which are frequently mentioned with a different context, and this is considered to be the false
Table 5.1: K-Mean cluster with K=2

<table>
<thead>
<tr>
<th>Cluster Number</th>
<th>Disease mentions</th>
<th>Unique disease names</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2533</td>
<td>351</td>
<td>13%</td>
</tr>
<tr>
<td>2</td>
<td>495</td>
<td>28</td>
<td>5%</td>
</tr>
</tbody>
</table>

Table 5.2: K-Mean cluster with K=3

<table>
<thead>
<tr>
<th>Cluster Number</th>
<th>Disease mentions</th>
<th>Unique disease names</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>872</td>
<td>207</td>
<td>23%</td>
</tr>
<tr>
<td>2</td>
<td>1662</td>
<td>279</td>
<td>16%</td>
</tr>
<tr>
<td>3</td>
<td>492</td>
<td>24</td>
<td>4%</td>
</tr>
</tbody>
</table>

Table 5.3: K-Mean cluster with K=4

<table>
<thead>
<tr>
<th>Cluster Number</th>
<th>Disease mentions</th>
<th>Unique disease names</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1320</td>
<td>290</td>
<td>21%</td>
</tr>
<tr>
<td>2</td>
<td>350</td>
<td>110</td>
<td>31%</td>
</tr>
<tr>
<td>3</td>
<td>865</td>
<td>141</td>
<td>16%</td>
</tr>
<tr>
<td>4</td>
<td>493</td>
<td>26</td>
<td>5%</td>
</tr>
</tbody>
</table>

Table 5.4: K-Mean cluster with K=5

<table>
<thead>
<tr>
<th>Cluster Number</th>
<th>Disease mentions</th>
<th>Unique disease names</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1292</td>
<td>289</td>
<td>22%</td>
</tr>
<tr>
<td>2</td>
<td>320</td>
<td>101</td>
<td>31%</td>
</tr>
<tr>
<td>3</td>
<td>348</td>
<td>109</td>
<td>31%</td>
</tr>
<tr>
<td>4</td>
<td>575</td>
<td>101</td>
<td>17%</td>
</tr>
<tr>
<td>5</td>
<td>493</td>
<td>26</td>
<td>5%</td>
</tr>
</tbody>
</table>

cluster. On the other hand, a high ratio means that the cluster contains a large amount of unique and real disease names, where only disease names are used because it is rare to find a specific disease name used frequently with a different context.

As shown in Figure 5.1, the red cluster contains almost 500 mentions with only 26 unique terms, which means this is the cluster that contains the few diseases that occur frequently and when they are used they usually do not refer to a disease. On the other hand, if we look at the green clusters we will see that they contain clear and specific disease names. Moreover, the other clusters are not used since they do not contain crucial information the could backup our second hypothesis because they might contain abbreviations or generic words.
For the final training, we chose the K-Mean cluster with K=4, because it gave us the highest accuracy on the training set, since it contains the second cluster with the highest ratio equal to 31%. Also, the fourth cluster with the lowest ratio equals 5%. Finally, we used all the disease mentions in the cluster with the highest ratio, and the cluster with the lowest ratio to train our supervised classifier. The cluster with the highest ratio was used as a positive cluster, which represents the real disease mentions, and the cluster with the lowest ratio was used as a negative cluster, which refers to non-diseases.

5.1 Testing process

NCBI testing disease corpus [45] contains 100 abstracts with 960 annotated disease name mentions, as our first approach to overcoming the limitations of multi-word mentions. All the multi-word disease name mentions were connected with the underscore _ symbol, such as the following case:

- Before: Clustering of missense mutations in the <category="Modifier">ataxia-
telangiectasia</category> gene in a <category="SpecificDisease">sporadic T-cell leukaemia</category>.

- After: Clustering of mutations in the ataxia-telangiectasia gene in a sporadic T-cell leukaemia.

The next step was lower-casing and removing the stop words. In the previous step, we ignored the stop words that exist in the disease ontologies like (IS) and (AND), after that we tokenized each abstract, then we removed all the punctuation.

Then, we generated BERT vectors for all the input words by using the whole abstract. Finally, our model was used to classify if the input words are diseases or not. Where Class 1 is the positive prediction it represents terms that are classified as diseases. On the other hand, Class 2 is the predicted negatives that represent terms that are classified as non-diseases. The testing procedure is shown in Figure 5.2

![Figure 5.2: Testing procedure](image)

### 5.2 Evaluation

To evaluate the final model after the testing procedure was made, the binary classifier model had two classes; Class 1 is the predicted diseases, and Class 2 is the predicted non-diseases. The model output four outcomes are listed below:

- True Positives (TP): terms labeled as positive that are positive disease names
- False Positives (FP): terms labeled as positive that are negative non-disease names

- True Negatives (TN): terms labeled as negative that are negative non-disease names

- False Negatives (FN): terms labeled as negative that are actually positive disease names

The final model results are shown in Table 5.5, and these results were calculated as follows [51]:

\[ \text{Precision (P)} = \frac{TP}{TP + FP} \]  
\[ \text{Recall (R)} = \frac{TP}{TP + FN} \]  
\[ F1 - \text{score} = 2 \frac{P \times R}{P + R} \]  
\[ \text{Macro - Averaged Precision} = \frac{\text{Precision}_{\text{Class1}} + \text{Precision}_{\text{Class2}}}{2} \]  
\[ \text{Macro - Averaged Recall} = \frac{\text{Recall}_{\text{Class1}} + \text{Recall}_{\text{Class2}}}{2} \]  
\[ \text{Weighted - Averaged Precision} = \frac{P_{\text{Class1}} \times C1 + P_{\text{Class2}} \times C2}{C1 + C2} \]  
\[ \text{Weighted - Averaged Recall} = \frac{R_{\text{Class1}} \times C1 + R_{\text{Class2}} \times C2}{C1 + C2} \]

<table>
<thead>
<tr>
<th>Setup</th>
<th>Precision</th>
<th>Recall</th>
<th>F1-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class1</td>
<td>84%</td>
<td>68%</td>
<td>75%</td>
</tr>
<tr>
<td>Class2</td>
<td>97%</td>
<td>99%</td>
<td>98%</td>
</tr>
<tr>
<td>MacroAveraged</td>
<td>91%</td>
<td>84%</td>
<td>87%</td>
</tr>
<tr>
<td>WeightedAveraged</td>
<td>95%</td>
<td>96%</td>
<td>96%</td>
</tr>
</tbody>
</table>

Where C1 is the true number of disease names in the testing data set, and C2 is the true number of non-disease names in the testing data set, our model has a high accuracy rate of 96%. As our main task to design a named entity recognizer, Class
1 represents predicted disease names mentioned in the text. The precision of Class 1 was 84%, because it was affected by the false positives, where some terms were labeled as positive diseases, but they were in fact non-disease names. Two examples are given below:

- Mutations in a novel gene (WFS1) encoding a putative transmembrane protein were found in all affected individuals in six <category="Modifier">WFS</category> families, and these mutations were associated with the disease phenotype. WFS1 appears to function in the survival of islet beta-cells and neurons.

- <category="Modifier">Wilms tumor</category> 1 (WT1) genes are essential.

In the first case, WFS is an abbreviation used to refer to Wolfram syndrome, and the WFS mentioned in the text is annotated as a modifier in the test set. However, the other terms WFS1 are abbreviations used to refer to 'Wolfram syndrome type 1' and they are mentioned but not annotated by the human annotators as disease names. Therefore, they are considered as false positive. In the second case, the human annotator only annotated the Wilms tumor as a disease name but they didn’t consider the abbreviation used to refer to the disease name WT1, and when we found it we considered it as false positive.

Moreover, the recall result of 68% was highly affected by the false negatives, and terms labeled as negative, that are positive disease names, were annotated by the human annotator. They are negatively labeled as non-diseases, due to two reasons; either they do not exist in the disease ontologies or they are annotated as nested terms, as in this example:

- <category="Modifier">breast / ovarian cancer</category>

In this example, the annotator annotates 'breast / ovarian cancer' together, and it is impossible to find this match in the disease ontologies used in this model. Moreover,
most of the positive disease names are labeled as false negatives because of the term variations not present in the disease ontologies.

One of the main challenges in this study is to design a disease name recognizer, which can out-perform other lexical models. It is clear from Table 5.6 that our model outperforms the lexical models. Firstly, we compared our model against the full exact string matching lexical baseline. Secondly, we compared our model against the exact and approximate string matching lexical baseline, where we removed the punctuation as a preprocessing step.

The results shows that our model performed better than the other lexical approaches. These results were expected in lexical baselines, because the precision was affected by the ambiguous and nested terms. Also, lexical approaches do not consider the term variability.

<table>
<thead>
<tr>
<th>Setup</th>
<th>Precision</th>
<th>Recall</th>
<th>F-measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Without preprocessing</td>
<td>35%</td>
<td>61%</td>
<td>44%</td>
</tr>
<tr>
<td>With preprocessing</td>
<td>42%</td>
<td>68%</td>
<td>51%</td>
</tr>
<tr>
<td>Our Model</td>
<td>84%</td>
<td>68%</td>
<td>75%</td>
</tr>
</tbody>
</table>

Table 5.6: Performance compared against lexical baselines

The next evaluation that was done, was a comparison of our unsupervised model with another supervised model, called Banner [45]. As shown in Table 5.7, our unsupervised model shows a slightly better performance in precision. On the other hand, the performance is lower compared to the other measurement scores.

As a result, our unsupervised model still shows great potential, since the performance could be enhanced by many factors such as adding more training data or adding more filters.

<table>
<thead>
<tr>
<th>Approach</th>
<th>Setup</th>
<th>Precision</th>
<th>Recall</th>
<th>F-measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supervised</td>
<td>Banner</td>
<td>83%</td>
<td>80%</td>
<td>81%</td>
</tr>
<tr>
<td>Unsupervised</td>
<td>Our model</td>
<td>84%</td>
<td>68%</td>
<td>75%</td>
</tr>
</tbody>
</table>

Table 5.7: Performance compared against another supervised model
Chapter 6

Concluding Remarks

Diseases are important entities in biomedical research, and identifying mentions of diseases is essential to many text mining tasks in the biomedical research field [6]. To enable access to disease information, NER models are used, since they automatically identify the name of the biological entity mentioned in the text [3] in our case diseases. However, this is not an easy task because there are many difficulties that could arise such as the annotated mention not referring to a disease name.

The aim of this study is to design a name entity recognizer that can identify mentions of disease names in texts, and also to overcome the drawbacks of the other approaches such as lexical and supervised approaches. The main drawbacks of those approaches are that the performance is affected by the available amount of human-annotated data sets. Moreover, lexical approaches cannot distinguish between real mentions of diseases and mentions of other entities that share the same name or acronym.

To overcome the aforementioned problems, a novel model was designed to recognize disease mentions in the text, by using an unsupervised approach with several steps. Firstly, a lexical approach was used to extract the initial training set. Secondly, the initial training set was represented in vectors using Bert. Thirdly, the Bert vectors were clustered using K-mean clustering model. Finally, a supervised model was trained using the generated clusters from the previous step, and a filter was applied to further distinguish diseases and non-diseases.

The results of the model show a high accuracy rate of 96% and a high precision
rate of 84%. In addition to identifying disease terms, the model was able to identify what the non-diseases terms are with high precision and recall. Most of the disease terms were identified except for the nested terms and the terms that do exist in our dictionary due to several reasons such as the type of the disease.

Moreover, we are interested in expanding our model to include genes and drug ontologies, to annotate disease names, genes, and drugs. Furthermore, we plan to train and test our model using the Pubmed abstracts with more than one entity. Also, much more work is needed to improve the performance by adding additional string processing and more disease vocabularies.
REFERENCES


A Papers Submitted and Under Preparation