Exposé zur Diplomarbeit

Schnelles und genaues Erkennen von Proteinnamen in Texten

31.10.2009

Autorin : Alexandra Rostin

Email : alros@gmx.de

Betreuer : Prof. Dr. Ulf Leser

1. Hintergrund

Biomedizin ist ein sehr aktives Forschungsfeld, an dem sicherlich in Zukunft weiterhin intensiv geforscht werden wird. Widergespiegelt wird das in den – in großer Anzahl - schon vorhandenen sowie jedes Jahr neu hinzukommenden Publikationen. Um aus den vorhandenen Datenbeständen Erkenntnisse gewinnen zu können, müssen Biomediziner diese Informationsflut bewältigen. Diese scheint nur mit Hilfe computergestützter Verfahren beherrschbar zu sein.

Große Anteile biomedizinischer Daten liegen unstrukturiert als natürlichsprachlicher Text, beispielsweise als Artikel in einem Fachjournal, vor. In dieser Form lassen sich Daten nur schwer mittels maschinenbasierter Verfahren auswerten, da natürliche Sprache über eine zu komplexe Struktur verfügt. Dadurch sind in diesen Publikationen enthaltene Informationen in gewisser Hinsicht versteckt. Versteckte Informationen könnten beispielsweise die Erwähnung von Beziehungen zu anderen Genen oder Proteinen sein wie Protein-Protein-Interaktionen (PIR) oder die Erkennung eines synonymen Gennamens anhand der funktionellen Annotation eines Gens [LH05].

Immer größere Anteile biomedizinischer Daten liegen daher in strukturierter Form als biomedizinische Datenbank vor, um computergestützten Verfahren einen besseren Zugang zu gewähren. Durch das Hinzufügen von Metainformationen ermöglichen Datenbanken die Aufbereitung der Daten in einer - für eine bestimmte Aufgabe geeigneten - Form. Metadaten könnten beispielsweise die Autoren oder das Fachgebiet eines Artikels, die beschriebenen Gene und Interaktionen oder Verweise zu anderen Artikeln sein.

Die manuelle Erstellung und Pflege solcher Datenbanken ist sehr zeit- und kostenintensiv, daher gewinnt die automatische Erfassung von Publikationen mittels intelligenter computerbasierter Verfahren immer mehr an Bedeutung. Hierbei kommen Verfahren aus den Bereichen *Informationsextraktion* (IE), *Information Retrieval* (IR), *Textmining* (TM), *Machine Learning* (ML) und *Natural Language Processing* (NLP) zum Einsatz, um Informationen aus unstrukturierten Texten einer weiterführenden Datenverarbeitung zugänglich zu machen [FZKH05].

Einer der ersten Schritte einer solchen Verarbeitungskette ist in der Regel die Entdeckung von Entitäten anhand von Eigennamen (hier Gen- und Proteinnamen), die nachfolgend einer Klasse bzw. Kategorie zugeordnet werden, um weitere Bearbeitungsschritte zu ermöglichen.

Mit der Detektion von Eigennamen beschäftigt sich das *Named Entity Recognition* (NER) Problem. Ein System, dass dieses realisiert, wird NER-System genannt und besteht aus mehreren Verarbeitungsstufen. Diese erfüllen unterschiedliche Funktionen wie zum Beispiel die Zerlegung eines natürlichsprachlichen Textes in Token (Tokenizer) oder die Zuordnung eines jeden dieser Token zu einer Wortart (POS-Tagger). Die NER-Systeme können grob in wörterbuchbasiert (dictionary-based), regelbasiert (rule-based) und klassifikations-basiert (classification-based) unterteilt werden. Ein verwandtes Problem ist *Named Entity Normalization* (NEN), mit dem Ziel, Namen auf einem allgemein gültigen Identifikator abzubilden.

Die zur Zeit besten NER-Systeme zur Erkennung von Gen- und Protein-Namen benötigen aufgrund komplexer Textanalyse-Techniken viel Zeit, daher sind sie für den Einsatz auf sehr großen Korpora nicht geeignet.

Die Diplomarbeit findet im Rahmen des Projektes AliBaba[1] der Humboldt Universität statt (siehe [PNLH09]). Die Aufgabe von AliBaba ist es, in PubMed[2] gefundene Interaktionen bzw. Zusammenhänge zwischen Genen, Proteinen oder beidem, als Graphen zu visualisieren.

2. Zielstellung

Ziel dieser Diplomarbeit ist die Erforschung des TradeOffs zwischen Effizienz und Qualität von NER-Systemen zur Erkennung von Gen- und Proteinnamen. Die Fragestellungen lauten unter anderem:

- Welchen Einfluss haben Maßnahmen zur Reduzierung der Laufzeit auf die Qualität?

[1] http://alibaba.informatik.hu-berlin.de/

[2] http://www.ncbi.nlm.nih.gov/pubmed/ (biomedizinische Datenbank)

- Sind die präzisesten Verfahren immer diejenigen, die am aufwändigsten sind und somit eine lange Laufzeit haben, während die mit geringerer Laufzeit weniger präzise sind?
- Und würden sich diese so kombinieren lassen, dass am Ende ein Verfahren herauskommt, das Ergebnisse mit möglichst hoher Genauigkeit liefert und dessen Berechnung möglichst wenig Zeit kostet?

3. Herangehensweise

Der praktische Forschungsteil dieser Diplomarbeit gliedert sich in vier Schritte:

1. **Vorbereitung**

   Installation und Inbetriebnahme zur Verfügung stehender NER-Tools und Erstellung einer Evaluationsumgebung. Letzteres umfasst die Bestimmung von Korpora, die zur Zielstellung passen. In Frage kommt beispielsweise das *BioCreative gene mention task training set* (siehe [HBYV05]) und andere gebräuchliche Korpora[3].

2. **Vorstudie**

   Dient der Bestimmung von NER-Tools anhand einer ausführlichen Evaluation für die Hauptstudie. Ausgewählt wird aus den im vorherigen Schritt erfolgreich getesteten Tools. Auswahlkriterien könnten neben Performanz, Verfügbarkeit (als Sourcecode), Unterstützung von Schnittstellen oder Formaten, Erweiterbarkeit/Anpassbarkeit und einfacher Handhabung auch verwendetes NER-Verfahren sein. Einige mögliche Tools werden in 4.2. vorgestellt, die Vorstudie soll aber nicht auf diese beschränkt sein.

3. **Hauptstudie**

[3] unter http://biocreative.sourceforge.net/bio_corpora_links.html ist eine Sammlung biologischer Korpora zu finden

Beinhaltet die Bearbeitung der eigentlichen Fragestellung der Diplomarbeit - Auswirkungen von Reduzierungsmaßnahmen auf die Qualität der Ergebnisse anhand der in der Vorstudie gewählten Tools. Die Evaluation erfolgt über die im ersten Schritt ausgewählten Korpora.

4. Entwicklung eines NER-Tools

Als Abschluss des Forschungsteils soll eine technische Umsetzung des erfolgversprechendsten Verfahrens (möglichst niedrige Laufzeit und möglichst hohe Qualität) aus der Hauptstudie erfolgen. Dafür soll keine neue Software geschrieben werden, sondern aus Modifikationen, Erweiterungen und/oder Kombinationen der ausgewählten NER-Tools ein neues entstehen. Das so entstandene NER-Tool wird wiederum auf mehreren Korpora evaluiert.

4.NER-Verfahren und NER-Tools

Zuerst wird ein Überblick über unterschiedliche Verfahren für NER gegeben, danach ausgewählte Tools vorgestellt und diese in einer Tabelle gegenübergestellt. Als letztes werden Ideen vorgestellt, auf welcher Art und Weise eine Effizienzsteigerung erreicht werden könnte.

4.1. NER-Verfahren (allgemein)

NER-Verfahren lassen sich anhand verwendeter Methoden in drei Hauptkategorien unterteilen [LH05]:

wörterbuch-/lexikon-basiert

In diesem Verfahren werden Terme (Token) gegen eine Namensliste gematcht, ebenso werden Varianten der Schreibweise berücksichtigt. Verfahren dieser Kategorie haben zumeist eine hohe Precision, aber nur einen niedrigen Recall. Ein Vorteil ist, dass jedem Wörterbuch-Eintrag schon während der Erstellung ein allgemeingültiger Identifikator

zugeordnet werden kann und für NEN keine weiteren Schritte erforderlich sind. Nachteilig ist, dass unbekannte Namen nicht erkannt werden.

regel-basiert

Dieser Ansatz verwendet heuristische Regeln, die mit Hilfe von Expertenwissen erstellt wurden. Das ermöglicht das Trainieren auf kleinere Korpora, allerdings sind Verfahren dieser Kategorie weniger robust gegenüber ungesehenen Namen und ihre Erzeugung ist aufwändig. Anspruchsvoll erstellte regelbasierte Systeme verfügen meistens über eine sehr hohe Precision. Problematisch ist jedoch, dass mit steigender Precision der Recall immer kleiner wird, d.h. je spezifischer ein System ist, umso schlechter erkennt es andere Namen.

klassifikations-/Machine Learning (ML)-basiert.

Hierbei wird das NER-Problem auf ein Klassifikationsproblem abgebildet, wodurch der Einsatz unterschiedlichster ML-Techniken ermöglicht wird, wie *Naive Bayes* (NB), Entscheidungsbaum (*decision tree* - DT), *Support Vector Machine* (SVM), *Hidden Markov Model* (HMM) oder *Conditional Random Fields* (CRF). Wie bei dem regelbasierten Ansatz wird eine hohe Precision meistens zu Lasten des Recalls erreicht (siehe [FZH05] S.218, rechte Spalte).

In aktuellen Tools (ebenso bei den unter 4.2. beschriebenen) kommen hauptsächlich statistische ML-Verfahren wie CRF zum Einsatz, Wörterbücher werden häufig ergänzend verwendet.

4.2 Tools

Nachfolgend werden einige Tools genannt, die im Forschungsteil der Diplomarbeit verwendet werden könnten, aber nicht müssen. Ein wichtiger Aspekt bei der Auswahl war die Verfügbarkeit des Sourcecodes.

ABNER

Der Kern dieses NER-Tools besteht aus einem statistischen Machine Learning System (*linear-chain conditional random fields*) (siehe [S04]). Es wurde für *NLPBA/BioNLP2004 Shared Task Challenge*[4] von Burr Settles, Department of Computer Sciences, University of Wisconsin-Madison, entwickelt. Auf dem *BioCreative Korpus* erreicht es eine Precision von 74,5% und einen Recall von 65,9%. Die verwendete CRF-Komponente wurde mittels des NLP-Toolkit Mallet implementiert. ABNER wird sowohl als ausführbare Datei als auch als Quellcode angeboten [S05].

AliBaba

Die NER-Komponente AliBabas benutzt einen wörterbuch-basierten Ansatz, der in [KGRS05] beschrieben wird. Die Einträge bestehen aus regulären Ausdrücken, die aus verschiedenen bedeutenden biologischen Datenbanken gewonnen wurden z.B. UniProt, MesH, KEGG oder Grug-Bank. Der wörterbuch-basierte Ansatz wurde gewählt, damit zusätzlich Verweise zu externen Quellen gefunden werden, die wichtige Hinweise für eine Datenintegration liefern könnten (siehe [PNLH09]).

Banner

Banner wurde von Bob Leaman am BioAI lab[5] der Arizona State Univerisity als Open-Source-NER-Tool entwickelt und soll in erster Linie biomedizinischen Zwecken dienen. Auf dem BioCreative *Task II Gene Mentions* Korpus wurde eine Precision von 85,09% und ein Recall von 79,06% erreicht (siehe [LG08]). Verwendet werden statistische ML-Methoden (*Conditional Random Fields*). Ein wichtiger Punkt war es, ein möglichst wenig domänenspezifisches NER-Tool zu schaffen. Deshalb wurde auf die Verwendung von semantischen Eigenschaften oder regelbasierter Arbeitsschritte verzichtet. Außerdem war die Erweiterbarkeit ein wichtiger Aspekt.

[4] http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/ERtask/report.html
[5] http://www.fulton.asu.edu/~bioai/

GNAT

NEN-Tool, das Texte nach Gen-Erwähnungen (*Gen Mention Normalization* (GN) – siehe [HPLSG08]) sucht und diese auf EntrezGene[6] Identifikatoren abbildet. Für NER werden wörterbuch-basierte und statistische (CRF) Methoden verwendet. Durch Verwendung von Hintergrund-Informationen werden Namens-Ambiguitäten aufgelöst. GNAT enthält 3,5 Millionen Gene von rund 4700 Spezies (Stand September 2009). Zur Zeit besteht die Möglichkeit über ein Webinterface eine Anfrage zu stellen, allerdings ist der Zugriff beschränkt - es stehen nur 36.500 menschliche Gene (135.000 unterschiedliche Gennamen) zur Verfügung.

SciMiner

Webbasiertes NER-Tool zur Erkennung von Gen- und Proteinnamen mittels einer kontextspezifischen Analyse von MEDLINE Abstracts und Volltexten. Zum Einsatz kommt sowohl ein Wörterbuch sowie ein regelbasiertes Verfahren. Ambiguitäten (Mehrdeutigkeiten) werden mittels Scoring Schema, basierend auf Co-Occurence von Akronymen und übereinstimmender Beschreibungsterme, aufgelöst. Auf dem *BioCreAtIvE version 2 Gene Normalization Task* Korpus erreicht SciMiner eine Precision von 71,3% und einen Recall von 87,1% (siehe [HSSF09]). Die Verwendung eigener Filter und Korrektur des Ergebnisses durch den Benutzer wird unterstützt.

[6] Gen-Datenbank, http://www.ncbi.nlm.nih.gov/sites/entrez/?db=gene

Name	Zweck	Sprache	NER Technik	Verfügbarkeit	URL
ABNER	NER	Java	statistisches ML (CRF)	Sourcecode ausführbare Datei	http://pages.cs.wisc.edu/~bsettles/abner/
AliBaba	IE	--	Wörterbuch	Sourcecode (Lehrstuhl intern) Webinterface	http://bioinformatics.oxfordjournals.org/cgi/content/full/22/19/2444#B6 http://alibaba.informatik.hu-berlin.de/
Banner	NER	Java	statistisches ML (CRF)	Sourcecode	http://banner.sourceforge.net
GNAT	NEN	Java	statistisches ML (CRF)	ja (Lehrstuhl intern) Webinterface	http://cbioc.eas.asu.edu/gnat/
SciMiner	NER	Perl	Wörterbuch regelbasiert	Sourcecode ausführbare Datei Webinterface	http://jdrf.neurology.med.umich.edu/SciMiner/

Tabelle 1: Vergleich NER-Tools

4.3. Überlegungen zur Effizienzsteigerung

Unter diesem Punkt werden Vorschläge erläutert, wie eine Effizienzsteigerung für NER-Verfahren erreicht werden könnte. Grob lassen sich die Ideen danach trennen, ob ein einzelnes Verfahren angepasst oder mehrere miteinander kombiniert werden sollen. Unter 3. wird beispielhaft ein konkreter Ansatz beschrieben.

1. Betrachtung einzelner Verfahren

Dabei sind folgende Herangehensweisen denkbar:

1. Reduzierung der Feature-Menge bei ML-Verfahren

   Die Idee zu diesem Vorschlag kam durch Erfahrungen aus anderen ML-Experimenten. In diesen konnte beobachtet werden, dass sogar drastische Reduzierungen der Anzahl verwendeter Features nur zu geringen qualitativen Einbußen führten [H05].

2. Verwendung eines Indexes bei einem wörterbuch-basierten Verfahren

   Bei wörterbuch-basierten Verfahren könnte der Zugriff auf das Wörterbuch durch Benutzen eines Index effektiver gestaltet werden. Am weitesten verbreitet ist der Ansatz, NER-Wörterbücher als reguläre Automaten zu implementieren (zum Beispiel NER-System AliBabas). Dabei wird ein Term gegen alle anderen gematcht. Dadurch sind bestimmte Varianten wie z.B. Umsortieren von Namenselementen schwieriger zu berücksichtigen.

2. Kombination von Verfahren

Vorschläge unter diesem Punkt gehen davon aus, dass jedes Verfahren gewisse Stärken und Schwächen hat und diese durch eine geschickte Kombination ausgeglichen werden können. Folgende Kombinationsweisen wären unter anderem vorstellbar:

1. Optimierung von Recall / Precision

   ML-Verfahren lassen sich je nach Verwendungszweck auf unterschiedliche Kriterien/Gütemaße optimieren. Für eine Steigerung der Effizienz könnte der Umstand in Frage kommen, dass bei der Optimierung auf Precision oder Recall meist ein einfacheres Modell gefunden wird, welches kürzere Berechnungszeit ermöglichen sollte. Bei einer Kombination könnte dies ausgenutzt werden, indem erst ein Verfahren verwendet wird, dass einen hohen Recall hat und viele Treffer zurück liefert. Im nächsten Schritt wird dann die Relevanz der gefundenen Treffer überprüft, indem auf das Ergebnis ein auf Precision optimiertes Verfahren angewendet wird.

2. Filterung / Vorauswahl / 'Pipeline' von NER-Verfahren

   Es wird angenommen, dass mehrere NER-Verfahren hintereinander angewandt werden. Diese könnten sich von der Art (Wörterbuch, regelbasiert, ML-basiert) und/oder durch anders gewählte Optimierungen her unterscheiden. Leicht entscheidbare[7] Token werden dann in früheren und schwer entscheidbare in späteren Schritten klassifiziert. Eine

[7] darauf bezogen, dass nur wenige Eigenschaften berücksichtigt werden müssen

Reduzierung der Laufzeit soll dadurch erreicht werden, dass aufwändige, komplexe und rechenintensive Verfahren nur auf einem (möglichst kleinen) Teil der gesamten Token-Menge angewandt werden müssen.

3. Mehrheitsentscheid mehrerer einfacher Verfahren

   Denkbar wäre, dass jedes Verfahren eine bestimmte Eigenschaft überprüft oder aus den Regeln eines jeweils anderen Korpus abgeleitet bzw. trainiert wird. Ein Token (bzw. Phrase) gilt dann als Name, wenn eine bestimmte Anzahl z.B. über die Hälfte aller Verfahren dieses als Namen einstufen würden.

3. Kombination ML-Verfahren mit wörterbuch-basiertem Verfahren:

Zunächst wird ein ML-Verfahren, zum Beispiel HMM auf Recall optimiert, was zu einem einfacheren Modell und zu weniger Features, die berechnet werden müssen, führen sollte. Die damit gefundenen Namen bzw. Phrasen werden dann in einem Wörterbuch nachgeschlagen. Für dieses könnten wiederum laxere Kriterien gelten, als bei einem Wörterbuch, dass als alleiniges Verfahren benutzt wird.

4. Literaturquellen

[CH05] Cohen, A. M. und Hersh, W. R. (2005). *„A survey of current work in biomedical text mining."* Briefings in Bioinformatics 6(1): 57-71.

[FZKH05] Fluck, J., Zimmermann, M., Kurapkat, G. und Hofmann, M. (2005). *„Information extraction technologies for the life science industry."* Drug Discovery Today: Technologies 2(3): 217-224.

[H05] Hakenberg, J.(Corresponding author), Bickel, S., Plake, C., Brefeld, U., Zahn, H., Faulstich, L, Leser, U. und Scheffer, T. (2005) *„Systematic feature evaluation for gene name recognition"*, Report, BMC Bioinformatics, 6(Suppl I):S9, April 2005

[HBYV05] Hirschman,L, Yeh,A, Blaschke, C, und Valencia, A (2005) *„Overview of BioCreAtIvE: critical assessment of information extraction for biology"*, BMC Bioinformatics 2005, 6(Suppl 1):S1

[HPLSG08] Hakenberg, J., Plake, C., Leaman, R., Schroeder, M., Gonzalez, G. (2008) *„Inter-species normalization of gene mentions with GNAT"*, Bioinformatics, Vol. 24 ECCB 2008, S.i126–i132

[HSL07] Hakenberg, J., Schroeder, M., und Leser, U., *„Consensus pattern alignment to find protein-protein interactions in text."*, Proc. Second BioCreative Challenge Evaluation Workshop. Madrid, Spain (23-25 April 2007). ISBN 84-933255-6-2

[HSSF09] Hur,J., Schuyler, A., D., States, D. J., Feldman,E. L. (2009) *„SciMiner: Web-based literature mining tool for target identification and functional enrichment analysis"*, Bioinformatics Advance Access, February 2, 2009

[KGRS05] Kirsch, H., Gaudan, S. und Rebholz-Schuhmann,D. (2005) *„Distributed modules for text annotation and IE applied to the biomedical domain"*,International Journal of Medical Informatics

4. Literaturquellen

[KSS05] Kabiljo, R, Stoycheva, D und Shepherd, A. J. (2005) *„ProSpecTome: a new tagged corpus for protein named entity recognition“*, Oxford University Press

[LG08] Leaman, R. und Gonzalez, G. (2008). *"BANNER: an executable survey of advances in biomedical named entity recognition."* Pac Symp Biocomput: 652-63.

[LH05] Leser, U. und Hakenberg, J. (2005). *"What Makes a Gene Name? Named Entity Recognition in the Biomedical Literature."* Briefings in Bioinformatics 6(4): 357-369.

[PNLH09] Palaga, P., Nguyen, L., Leser, U. und Hakenberg, J. (2009) *„High-Performance Information Extraction with AliBaba.“*, Extending Database Technology (EDBT), St. Petersburg, Russia.

[S04] Settles, B., (2004) *„Biomedical Named Entity Recognition Using Conditional Random Fields and Rich Feature Sets“*, erschienen in *„Proceedings of the COLING 2004 International Joint Workshop on Natural Language Processing in Biomedicine and its Applications (NLPBA).“*, Genf, Schweiz, (2004)

[S05] Settles, B., (2004) „ABNER: an open source tool for automatically tagging genes, proteins and other entity names in text“, Bioinformatics, Vol. 21 no. 14 2005, pages 3191–3192