Maschinelle Sprachverarbeitung

Named Entity Recognition

Ulf Leser

Content of this Lecture

- Named Entity Recognition
  - Dictionary-based approaches
  - Rule-based approaches
  - ML-based approaches
- Named Entity Normalization
- Case studies

Information Extraction: What we need to do

Z-100 is an arabinomannan extracted from Mycobacterium tuberculosis that has various immunomodulatory activities, such as the induction of interleukin 12, interferon gamma (IFN-gamma) and beta-chemokines. The effects of Z-100 on human immunodeficiency virus type 1 (HIV-1) replication in human monocyte-derived macrophages (MDMs) are investigated in this paper. In MDMs, Z-100 markedly suppressed the replication of not only macrophage-tropic (M-tropic) HIV-1 strain (HIV-1JR-CSF), but also HIV-1 pseudotypes that possessed amphotropic Moloney murine leukemia virus or vesicular stomatitis virus G envelopes. Z-100 was found to inhibit HIV-1 expression, even when added 24 h after infection. In addition, it substantially inhibited the expression of the pNL43lucDeltaenv vector (in which the env gene is defective and the nef gene is replaced with the firefly luciferase gene) when this vector was transfected directly into MDMs. These findings suggest that Z-100 inhibits virus replication, mainly at HIV-1 transcription. However, Z-100 also downregulated expression of the cell surface receptors CD4 and CCR5 in MDMs, suggesting some inhibitory effect on HIV-1 entry. Further experiments revealed that Z-100 induced IFN-beta production in these cells, resulting in induction of the 16-kDa CCAAT/enhancer binding protein (C/EBP) beta transcription factor that represses HIV-1 long terminal repeat transcription. These effects were alleviated by SB 203580, a specific inhibitor of p38 mitogen-activated protein kinases (MAPK), indicating that the p38 MAPK signalling pathway was involved in Z-100-induced repression of HIV-1 replication in MDMs. These findings suggest that Z-100 might be a useful immunomodulator for control of HIV-1 infection.

Find Entity Names (Multiple Classes)

Z-100 is an ***arabinomannan*** extracted from Mycobacterium tuberculosis that has various immunomodulatory activities, such as the induction of **interleukin 12**, **interferon gamma** (**IFN-gamma**) and beta-chemokines. The effects of ***Z-100*** on human immunodeficiency virus type 1 (HIV-1) replication in human **monocyte-derived macrophages** (**MDMs**) are investigated in this paper. In **MDMs**, ***Z-100*** markedly suppressed the replication of not only macrophage-tropic (M-tropic) HIV-1 strain (HIV-1JR-CSF), but also HIV-1 pseudotypes that possessed amphotropic Moloney murine leukemia virus or vesicular stomatitis virus G envelopes. ***Z-100*** was found to inhibit HIV-1 expression, even when added 24 h after infection. In addition, it substantially inhibited the expression of the pNL43lucDeltaenv vector (in which the ***env*** gene is defective and the ***nef*** gene is replaced with the ***firefly luciferase*** gene) when this vector was transfected directly into **MDMs**. These findings suggest that ***Z-100*** inhibits virus replication, mainly at HIV-1 transcription. However, ***Z-100*** also downregulated expression of the cell surface receptors **CD4** and **CCR5** in **MDMs**, suggesting some inhibitory effect on HIV-1 entry. Further experiments revealed that ***Z-100*** induced **IFN-beta** production in these cells, resulting in induction of the 16-kDa **CCAAT/enhancer binding protein** (**C/EBP**) **beta transcription factor** that represses HIV-1 long terminal repeat transcription. These effects were alleviated by SB 203580, a specific inhibitor of **p38 mitogen-activated protein kinases** (**MAPK**), indicating that the **p38 MAPK** signalling pathway was involved in ***Z-100***-induced repression of HIV-1 replication in **MDMs**. These findings suggest that ***Z-100*** might be a useful immunomodulator for control of HIV-1 infection.

Normalizing Entity Names

Tax-ID: 9015

CHembl-ID: 123335225

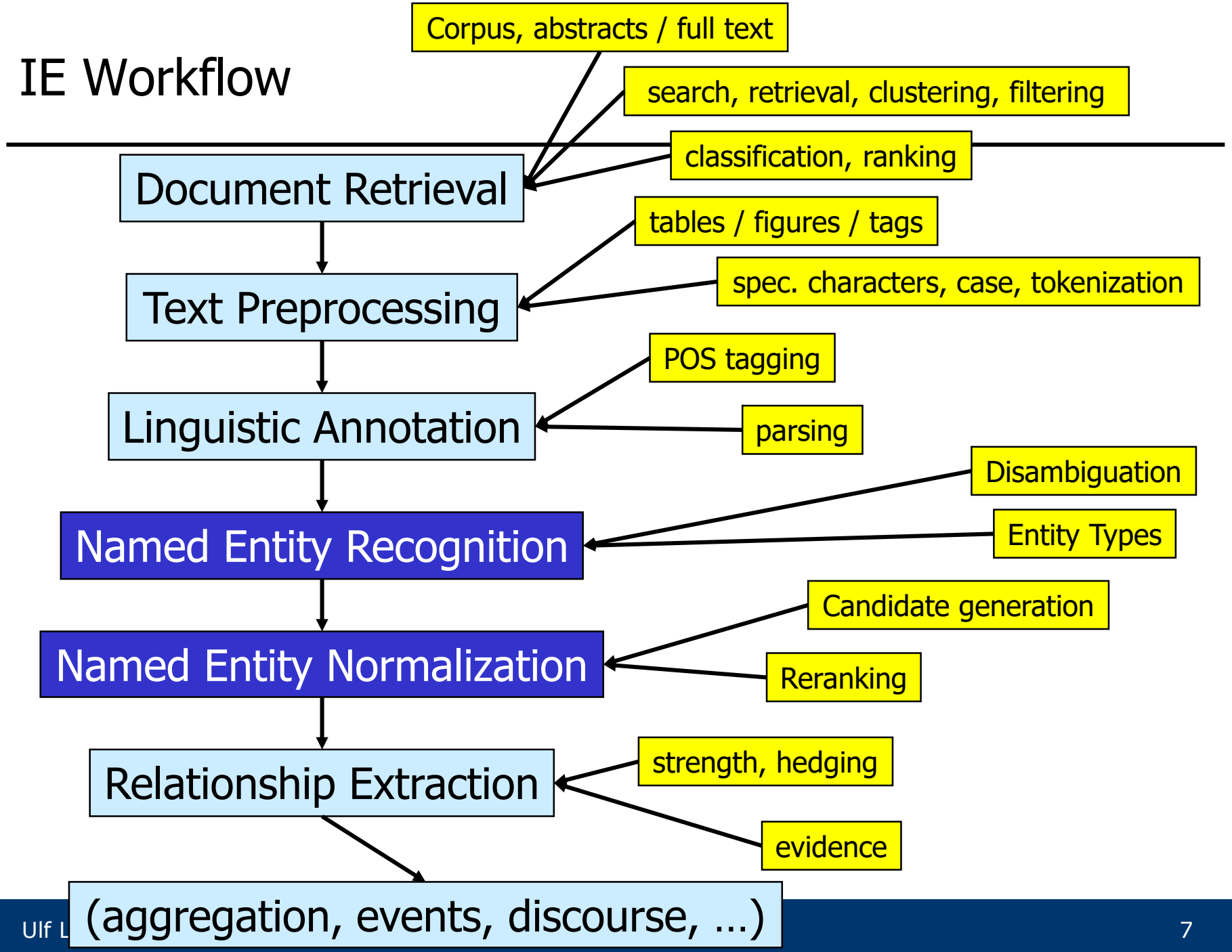
RefSeq: rf9876s25225

Z-100 is an ***arabinomannan*** extracted from Mycobacterium tuberculosis that has various immunomodulatory activities, such as the induction of **interleukin 12**, **interferon gamma** (**IFN-gamma**) and beta-chemokines. The effects of ***Z-100*** on human immunodeficiency virus type 1 (HIV-1) replication in human **monocyte-derived macrophages** (**MDMs**) are investigated in this paper. In **MDMs**, ***Z-100*** markedly suppressed the replication of not only macrophage-tropic (M-tropic) HIV-1 strain (HIV pes that possessed amphotropic Moloney murine leukemia v velopes. ***Z-100*** was found to inhibit HIV-1 expression, even when added 24 h after infection. In addition, it substantially inhibited the expression of the pNL43lucDeltaenv vector (in which the ***env*** gene is defective and the ***nef*** gene is replaced with the ***firefly luciferase*** gene) when this vector was transfected directly into **MDMs**. These findings suggest that ***Z-100*** inhibits virus replication, mainly at HIV-1 transcription. However, ***Z-100*** also downregulated expression of the cell surface receptors **CD4** and **CCR5** in **MDMs**, suggesting some inhibitory effect on HIV-1 entry. Further experiments revealed that ***Z-100*** induced **IFN-beta** production in these cells, resulting in induction of the 16-kDa **CCAAT/enhancer binding protein** (**C/EBP**) **beta transcription factor** that represses HIV-1 long terminal repeat transcription. These effects were alleviated by SB 203580 ctivated protein kinases (**MAPK**), indicating that the **p38 MA** in ***Z-100***-induced repression of HIV-1 replication in **MDMs**. These findings suggest that ***Z-100*** might be a useful immunomodulator for control of HIV-1 infection.

Relationship Extraction (RE)

Z-100 is an *arabinomannan* extracted from Mycobacterium tuberculosis that has various immunomodulatory activities, such as the induction of **interleukin 12**, **interferon gamma** (**IFN-gamma**) and beta-chemokines. The effects of *Z-100* on human immunodeficiency virus type 1 (HIV-1) replication in human **monocyte-derived macrophages** (**MDMs**) are investigated in this paper. In **MDMs**, *Z-100* markedly suppressed the replication of not only macrophage-tropic (M-tropic) HIV-1 strain (HIV-1JR-CSF), but also HIV-1 pseudotypes that possessed amphotropic Moloney murine leukemia virus or vesicular stomatitis virus G envelopes. *Z-100* was found to inhibit HIV-1 expression, even when added 24 h after infection. In addition, it substantially inhibited the expression of the pNL43lucDeltaenv vector (in which the ***env*** gene is defective and the ***nef*** gene is replaced with the ***firefly luciferase*** gene) when this vector was transfected directly into **MDMs**. These findings suggest that *Z-100* inhibits virus replication, mainly at HIV-1 transcription. However, *Z-100* also downregulated expression of the cell surface receptors **CD4** and **CCR5** in **MDMs**, suggesting some inhibitory effect on HIV-1 entry. Further experiments revealed that *Z-100* induced **IFN-beta** production in these cells, resulting in induction of the 16-kDa **CCAAT/enhancer binding protein** (**C/EBP**) **beta transcription factor** that represses HIV-1 long terminal repeat transcription. These effects were alleviated by SB 203580, a specific inhibitor of **p38 mitogen-activated protein kinases** (**MAPK**), indicating that the **p38 MAPK** signalling pathway was involved in *Z-100*-induced repression of HIV-1 replication in **MDMs**. These findings suggest that *Z-100* might be a useful immunomodulator for control of HIV-1 infection.

IE Workflow


Corpus, abstracts / full text
search, retrieval, clustering, filtering
classification, ranking
Document Retrieval
tables / figures / tags
spec. characters, case, tokenization
Text Preprocessing
POS tagging
parsing
Linguistic Annotation
Disambiguation
Entity Types
Named Entity Recognition
Candidate generation
Reranking
Named Entity Normalization
strength, hedging
evidence
Relationship Extraction
(aggregation, events, discourse, ...)

Named Entity Recognition (NER)

- Task: Find all mentions of a given type of entities in a text
  - Genes, diseases, companies, persons, parties, …
  - Different levels of granularity: Molecular entities, genes, mRNA, exons, human genes, genes implicated in cancer, …
  - Entities with a fuzzy definition: Earthquakes, symptoms, temporal expressions, relative directions, …
- Difficulties
  - Complete set of all existing entities often not known
  - Spelling variations and spelling errors
  - Entity names may span more than one token (also non-continuous)
  - Homonyms: Same tokens, different meaning
- Does usually not include referential mentions
  - Anaphora resolution (can be solved by classification)

Examples

- High plasma AVP levels observed in the two cases suggest that SSRIs stimulate AVP secretion, thereby causing SIADH
- A Drosophila shc gene product is implicated in signaling by the DER receptor tyrosine kinase.
- The human T cell leukemia lymphotropic virus type 1 Tax protein represses MyoD-dependent transcription by inhibiting MyoD-binding to the KIX domain of p300.
- The tumor necrosis factor alpha and dkfzp779b086 bind to the human mono-adp-ribosyltransferase.

Examples

- High plasma AVP levels observed in the two cases suggest that SSRIs stimulate AVP secretion, thereby causing SIADH
  - Requires domain knowledge
- A Drosophila shc gene product is implicated in signaling by the DER receptor tyrosine kinase.
  - Has to deal with ambiguities (context is important)
- The human T cell leukemia lymphotropic virus type 1 Tax protein represses MyoD-dependent transcription by inhibiting MyoD-binding to the KIX domain of p300.
  - Often has no clear answer (borders)
- The tumor necrosis factor alpha and dkfzp779b086 bind to the human mono-adp-ribosyltransferase.
  - May use very specific words or consist of rather common words

Some Funny Gene Names

- Dickkopf, zerknüllt, Spätzle
- a (Entrez Gene 43852)
- Lush (40136); (Protein mediates responses to alcohols)
- Van gogh (35922) (Have swirling wing-hair patterns)
- Wish
- Soul
- the
- …
- Obviously, all of these are homonyms

Abbreviations

- ACE
  - angiotensin converting enzyme
  - affinity capillary electrophoresis
  - Acetylcholinesterase
  - ACE I, a nephrotoxic drug
  - Anevrysme de l'aorte abdominale
  - acetosyringone
  - Addenbrooke's cognitive examination
  - Dirección Médica de Fundació ACE
  - …
- >60 definitions for ACE in Wikipedia
- Study says: 80% of all acronyms in Medline are not unique

Related: Word Sense Disambiguation (WSD)

- Often, terms can have multiple meanings / senses
  - Bass can be a fish or an instrument
- WSD: Assign the correct sense to a term (or all terms) in a given text
  - Set of senses: Language dictionaries
  - Related problem: Word Sense Discovery (find all existing senses)
  - Needs to consider the context of the term mention
    - "Can you play the bass?"
- Polysemy: Senses that are very close to each other
  - "The company Thomas Cook was named after Thomas Cook"

Single / Multi-Class NER

- Single-class NER: Recognize terms of one particular class
  - E.g.: Genes, diseases
  - Like WSD with two possible senses: The class and "other"
- Multi-class NER: Recognize terms of multiple classes and assign the correct class
  - E.g.: Genes, diseases, species, and tissues
  - Like WSD with two k+1 possible senses: The k classes and "other"
- Note: Finding the start and end of terms is part of NER

- Named Entity Normalization: Disambiguate entities of the same class into their individual instances

Content of this Lecture

- Named Entity Recognition
  - Dictionary-based approaches
  - Rule-based approaches
  - ML-based approaches
- Named Entity Normalization
- Case studies

Dictionary-Based NER

- Gazetteer or dictionary
  - A gazetteer originally is a list of geographic names with locations
  - In information extraction, a gazetteer is a list of names
- Dictionary-based NER (for single token entities)
  - Obtain a dictionary of all names of entities you are interested in
    - Dictionary should include all synonyms
  - Match every token in the text against the dictionary
  - Exact matching: Only find occurrences exactly matching a dictionary entry
  - Similarity-based matching: Also find (slight) variations

Dynamic Domains

- Can we always build a dictionary of all entities of a class?
  - Finding all street names in Berlin is relatively simple
  - Finding all geographic locations is more difficult
    - Places, buildings, hills, woods, …
  - Finding all person names in Germany is even more difficult
    - New persons are born all the time
      - Mostly new combinations of known first / last names
    - New names immigrate all the time
    - Other languages are much more innovative with names (initials, J.R: junior, Schewarnadze (son), Saakaschwili (child), Hadschi Halef Omar Ben Hadschi Abul Abbas Ibn Hadschi Dawuhd al Gossarah, …
  - Finding all company names is even more difficult
    - Companies are created and closed all the time
    - No real naming conventions (Remember the ".com" phase)
    - Often with fixed elements (GmbH, AG, inc., …)

Funny First Names [Berliner Zeitung, 2008]

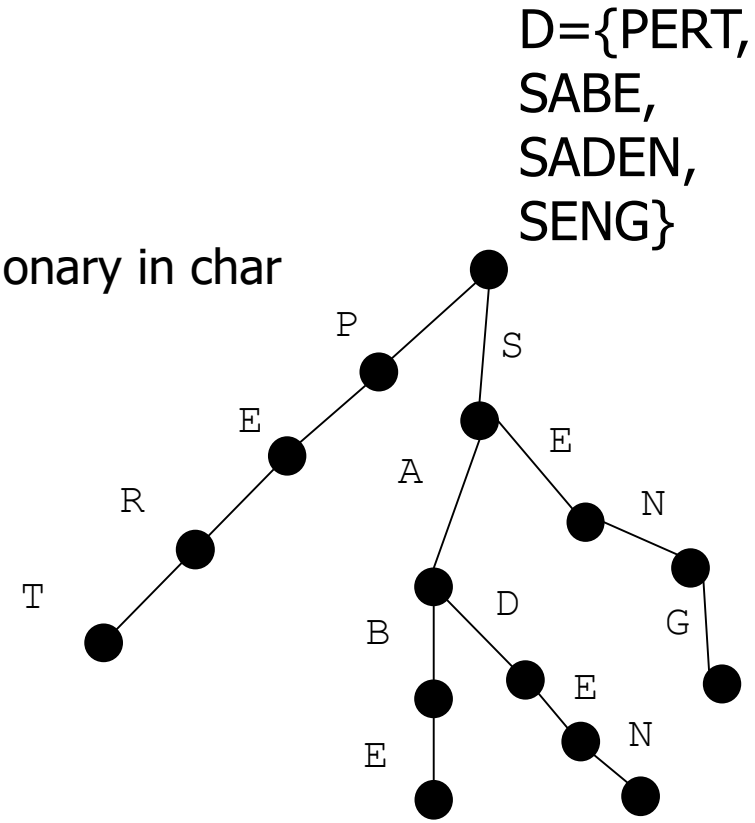
- Regulations in Germany: „Die Schreibweise ist den Regeln der Rechtschreibung unterworfen. Biblische Namen mit negativer Assoziation wie Judas oder Kain sind nicht erlaubt, ebenso wenig Markennamen, die nicht mit Vornamen identisch sind, Adelstitel, Orts- und Städtenamen. Also nichts mit Arizona, Sierra Nevada oder Schweinfurt. Ausnahmen wie Mercedes, Paris und San Diego bestätigen allerdings die Regel. Außerdem muss der Vorname das Geschlecht erkennen lassen, weshalb ein Kind namens Kim einen zweiten Vornamen braucht."
  - Internationale Promis hätten in Deutschland schlechte Karten. Ist der Name von Nicole Kidmans Tochter Sunday Rose weiblich? Nein, der Sonntag ist so männlich wie Freitag aus Robinson Crusoe. Und was ist mit Gwyneth Paltrows Tochter Apple? Im Deutschen wäre es der Apfel ... Da wir schon mal beim Obst sind: Eine Lehrerin in Neuseeland heißt Cherry. Kirsche. Immerhin: die Kirsche. Auch viele Frauen namens Fern gibt es im Land des Silberfarns. Und ganz im Trend der handy- und SMS-süchtigen jungen Generation kamen im vergangenen Jahr reichlich Knaben namens JJ, C, CJ, T, TJ und AJ auf die Welt. Die weibliche Antwort darauf ist Tequila. Zur besseren Verdauung aller schwer verdaulichen Vornamen.
- Genehmigt: Pepsi-Carola, Napoleon, Rasputin, Rapunzel, Sunshine, Sonne
- Abgelehnt: Möwe, Porsche, Pfefferminze, Lenin, Crazy Horse, Störenfried

Example: Finding all gene names is hard

- New genes are found or genes are re-discovered all the time
- Definition of a gene is not clear at all (splicing, miRNA, …)
- Disambiguation between gene, transcripts, and encoded proteins often almost impossible
- No (successful) naming convention
  - Discoverer, disease, location, phenotype, species, cell type, …
- Much "legacy" text which is only a couple of years old
- Frequent use of abbreviations
- Often multiple tokens long
- Much usage of special characters ("-", "", "/", digits, …)
- Use of common English words (hedgehog, Dickkopf, soul, …)
- Rediscoveries lead to multiple names for same gene
- "Same" gene exist in multiple species
- "Same" gene may appear at multiple locations in a genome
- Pseudo-genes
- …

Dictionary-Based NER: Exact Matching

- Algorithmically simple
- Without tokenization
  - Use Aho-Corasick algorithm
  - Solves the problem in O(m+|D|)
    - m: text length in char, |D|: Size of dictionary in char
- With tokenized input text
  - Classical dictionary problem in computer science
  - Best: Build hash table for dictionary
  - With suitable hash function, this achieves O(|t|) runtime in practice
    - t: Length of input token
  - Alternative: Sort dictionary and use binsearch



Pro / Contra

- Fast, but low quality

- Precision impacted by lack of disambiguation when entity names are ambiguous (THE)
- Low recall
  - Spelling variations
  - Multi-token entities: Token need not appear all and in given order
- No extrapolation towards "typical" entity names

Similarity-Based Dictionary NER

- Even in static areas, names need not appear exactly
  - Yahoo, yahoo, Yahoo!, yahoo.com, yaho (typo), …
  - Die Geissens, die Geissen's, die Geissen`s, die Geißens, …
- Solution: Similarity-based matching
  - Consider as an entity every (set of) token that is similar to an entry in the dictionary
  - Single term may produce multiple and different matches – named entity normalization
  - Similarity must be defined
    - Liberal measure / threshold: High recall, low precision
    - Strict measure / threshold: Lower recall, higher precision
    - (Individual) thresholds can be learned
  - Still disregards context – impact on precision

Implementation

- Approach 1: Generate a "fuzzified" dictionary
  - Rewrite every entry to generate common synonyms
    - Plural s, genitive s, upper / lower case, …
  - Apply exact matching
  - Fast, better recall than exact match, but only basic spelling variants
- Approach 2: Compare every token to every dictionary entry
  - Compute similarity, accept if threshold is passed
  - Slow, more flexibility regarding precision/recall trade-off
  - Depending on similarity function, dictionary can be indexed
  - Good similarity functions are domain-specific
    - Person names: Punish special characters, ignore case
    - Gene names: Reward spec.char. + digit (THE-3, THE'3), retain case

Popular: Edit Distance, Levenshtein distance

- Compute the minimal number of edit operations needed to transform token t into entry e
  - Typical operations: character insertion, deletion, replacement
  - Requires O(|t|*|e|) operations – very slow
- Should be length-normalized
  - Tor-Kur, Schifffahrt-Schifffahrten (distance 2)
- Should use different weights for different characters
  - Meier – Maier, Tobel – Hobel (distance 1)
- Works best for rather long entity names
- Ineffective for short names (e.g. abbreviations)
  - "operation" not fine-grained enough
- Much research in efficient index structures

Popular: Jaccard Distance, Dice Coefficient

- For a given k, let E be the set of k-grams of e and T the set of k-grams of t:

$$Jaccard(t,e) = \frac{|T \cap E|}{|T \cup E|} \qquad Dice(t,e) = \frac{2 * |T \cap E|}{|T| + |E|}$$

- Properties
  - Fast: Can be computed in O(|E|+|T|)
    - Assuming precomputed, sorted E and T
    - K-grams can be considered as tokens: Use inverted indexes
  - Differences in start / end of token count less than diffs in middle
  - Large k improve precision and speed, small values improve recall
    - Relative to length of terms / entries – must be tuned
    - Very large k: ~ exact matching; very low k: ~ character distribution
    - Often multiple k are used simultaneously
  - Can be used to derive lower bound for edit distance

Similarity Measures

List of string metrics [edit]

- Levenshtein distance
- Damerau–Levenshtein distance
- Sørensen–Dice coefficient
- Block distance or L1 distance or City block distance
- Jaro–Winkler distance
- Simple matching coefficient (SMC)
- Jaccard similarity or Jaccard coefficient or Tanimoto coefficient
- Tversky index
- Overlap coefficient
- Variational distance
- Hellinger distance or Bhattacharyya distance
- Information radius (Jensen–Shannon divergence)
- Skew divergence
- Confusion probability
- Tau metric, an approximation of the Kullback–Leibler divergence
- Fellegi and Sunters metric (SFS)
- Maximal matches
- Grammar-based distance
- TFIDF distance metric[3]

Source: Wikipedia

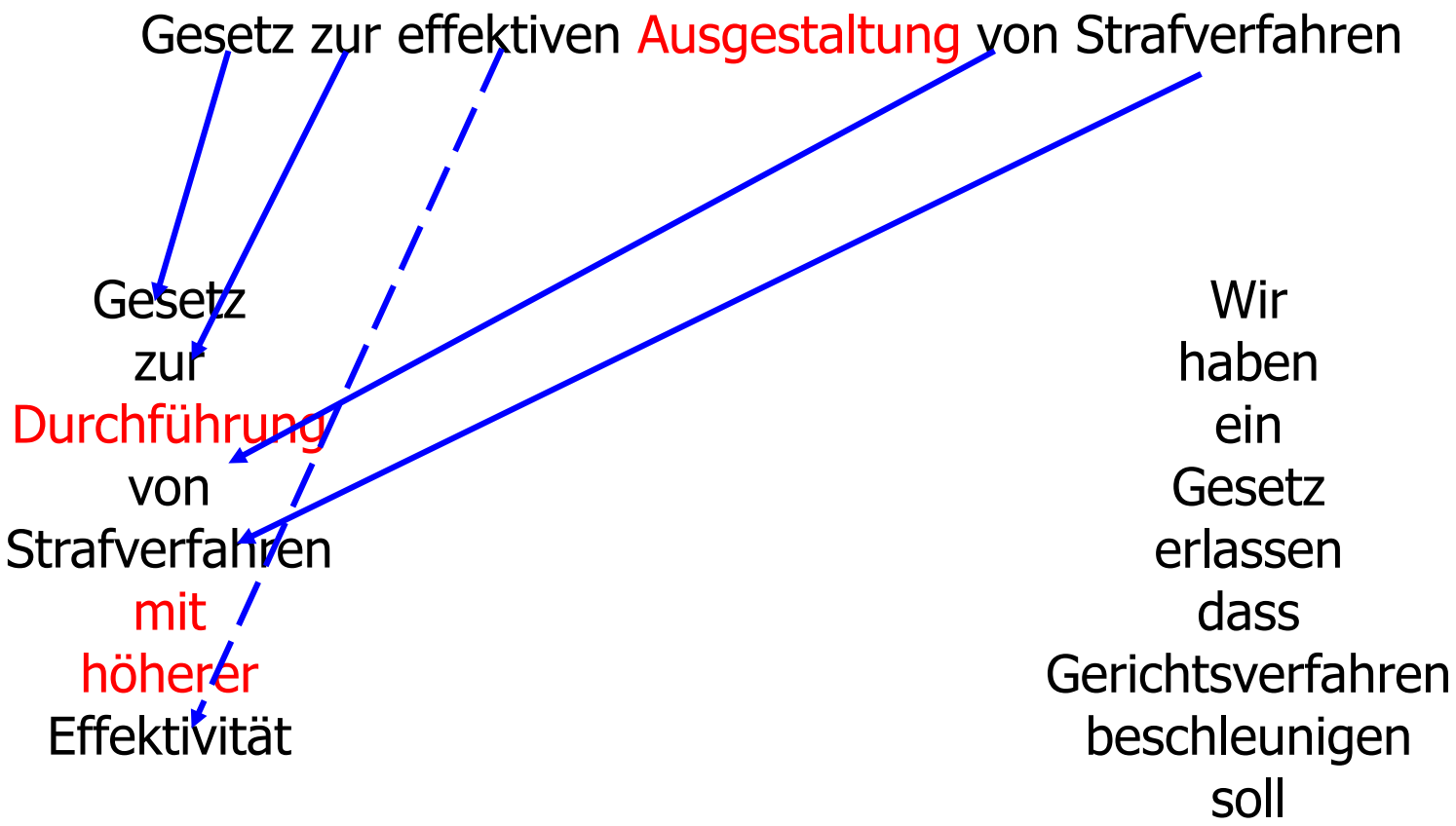
Dictionary-Based NER: Multiple Token

- Entities may consist of multiple entities
  - "Gesetz zur effektiveren und praxistauglicheren Ausgestaltung des Strafverfahrens"
  - "Gesetz zur effektiveren und praxistauglicheren Ausgestaltung von Strafverfahren"
  - "Gesetz zur effektiven Ausgestaltung von Strafverfahren"
  - „Wir haben ein Gesetz erlassen, dass Gerichtsverfahren beschleunigen soll"
- Very long entities come close to topical phrase classification
  - GO terms: "*Negative regulation of anterior neural cell fate commitment of the neural plate by fibroblast growth factor receptor signaling pathway*"
- Creates special problem during NER
  - Token of entity may be missing and new token may appear
  - Token may appear in different order
  - Token may be replaced by other token with same/similar meaning
- In principle, we need to match all token of a potential occurrence in a text with all token of the respective entry
  - Using single-token similarity methods

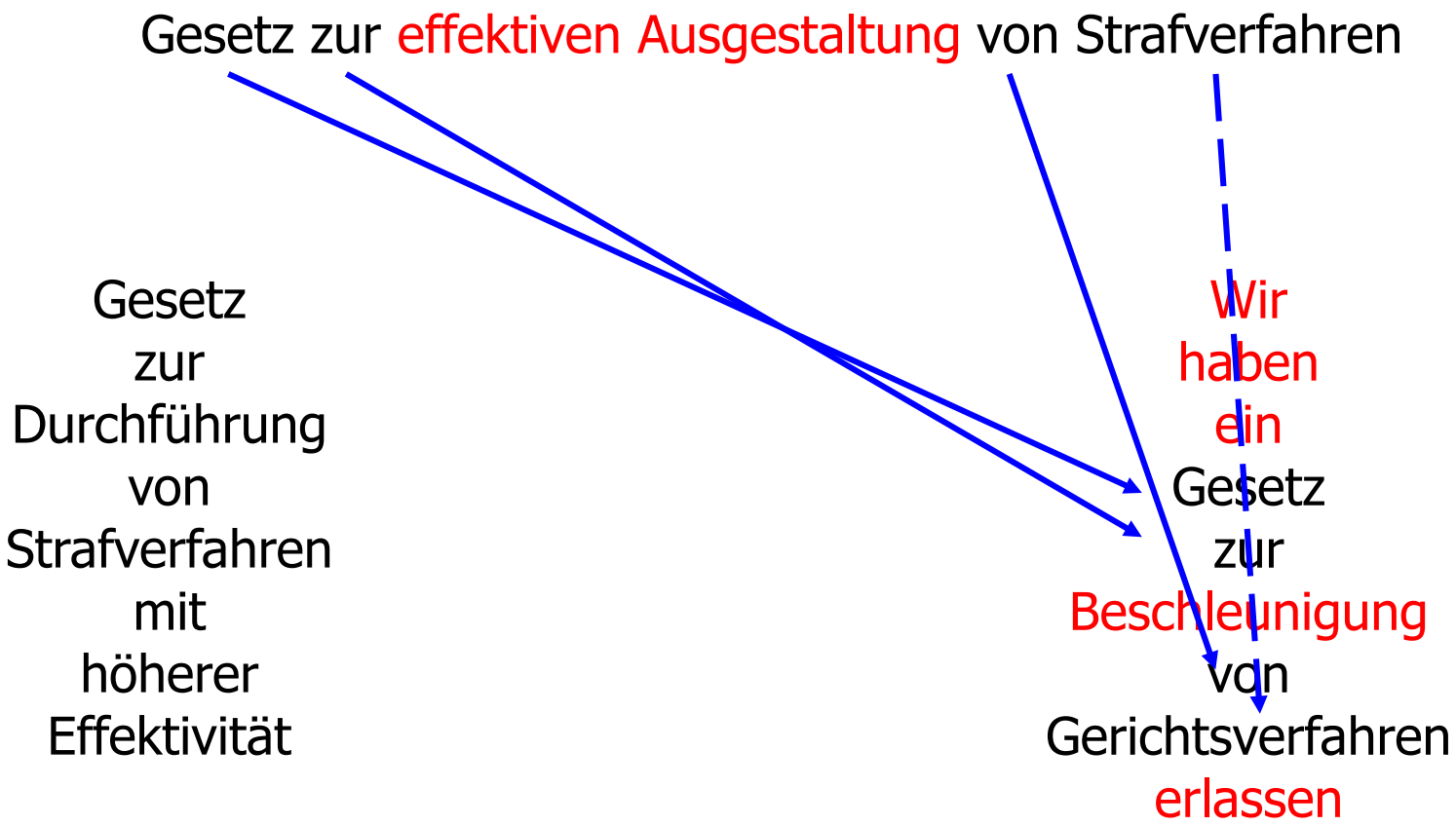
Algorithmic framework

- Compute similarity of all text token with all dictionary token
  - Consider all individual token of the dictionary, not only all entries
  - Speed-up: Remove all similarities below a given threshold
- Move a sliding window over text
  - Window length: Difficult! Length of longest dict entry plus a bit?
- For every entry / window pair
  - Compute bipartite matching of text token with entry token
    - Assumption: Entry token cannot generate multiple text token
    - This can be tricky if token have multiple potential matches
    - Bipartite matching: $O(n^3)$ (n = length of window = length of entry)
  - Compute an aggregated score
- Return entry with highest score for this window
  - Or nothing if threshold not met

Example


Gesetz zur effektiven Ausgestaltung von Strafverfahren
Gesetz zur Durchführung von Strafverfahren mit höherer Effektivität
Wir haben ein Gesetz erlassen dass Gerichtsverfahren beschleunigen soll

Example


Gesetz zur effektiven Ausgestaltung von Strafverfahren
Gesetz zur Durchführung von Strafverfahren mit höherer Effektivität
Wir haben ein Gesetz zur Beschleunigung von Gerichtsverfahren erlassen

Aggregated Scores

- Typical ingredients
  - Similarity of token pairs in the optimal matching
  - tfidf value of matched token in window (and in entry?)
  - Number of unmatched token in both window and entry
  - Difference in order of token, distance of matching pairs
- Example (T, E: text/entry; t/e: token of T/E; (t,e): matched token pair)

$$token(T,E) = \sum_{all\ matches\ (t,e)} sim(t,e) * tfidf(t) - \sum_{unmatched\ s\ in\ T\ or\ in\ E} tfidf(s)$$

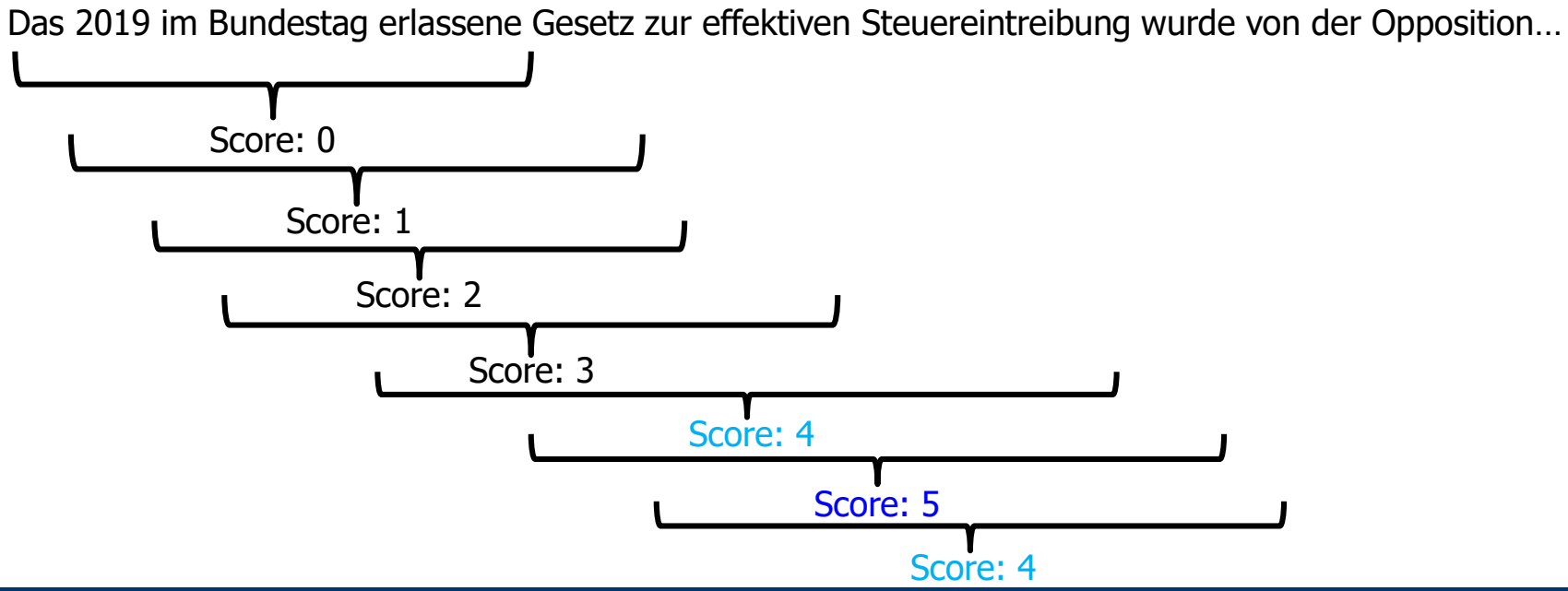
$$order(T,E) = \frac{\left|\left\{\left((t_i,e_k),(t_j,e_l)\right) \mid i<j\ and\ k<l\right\}\right|}{\left|\left\{\left((t_i,e_k),(t_j,e_l)\right) \mid i<j\right\}\right|}$$

$$agg(T,E) = token(T,E) * order(E,T)^{-1}$$

Disambiguate Overlapping Windows

- Especially entries with many tokens often produce overlapping matches

Gesetz zur effektiven Steuereintreibung



Disambiguate Overlapping Windows

- Simple solution: If entry E matches in overlapping windows, only keep highest scoring match
- Things get tricky when different entries produce matches over overlapping windows
- Things get tricky when multiple optima exist
- …

Properties of Dictionary-Based NER

- Advantages: Simple, fast, naturally includes NEN
  - Typical baseline system
  - Easiest solution, lay persons use it as synonym for NER
  - If entity names typically are short: Much faster than machine learning or rule-based systems
- Well suited for static (closed) classes with few entities
  - Find Nobel price winners; find US presidents; …
  - Problems remain: Ambiguous names
- For dynamic classes
  - Performance depends on dictionary size, level of ambiguity, and similarity function
  - Usually one expects high precision at low recall
    - No abstraction of entries into properties of "typical" entity names

Content of this Lecture

- Named Entity Recognition
  - Dictionary-based approaches
  - Rule-based approaches
  - ML-based approaches
- Named Entity Normalization
- Case studies

Rule-Based Systems

- Define rules that capture indications for of a NE
  - Combine context words, POS tags, surface properties, …
    - [PERSON] earns [MONEY] USD
    - [PERSON] join* [ORGANIZATION]
    - the [PROTEIN]/NNS receptor
- Labor-intensive: Someone must define many rules
- Typical trade-off
  - Long, precise rules: Very good precision, low recall
  - Short, general rules: Bad precision, good recall
- Often used in combination, e.g., use ML-based NER and rules for post-processing (filtering false positives)
- Somewhat old-fashioned, but …

Rule-Based or Machine-Learning-Based?

Rule-based Information Extraction is Dead!
Long Live Rule-based Information Extraction Systems!

Laura Chiticariu
IBM Research - Almaden
San Jose, CA
chiti@us.ibm.com

Yunyao Li
IBM Research - Almaden
San Jose, CA
yunyaoli@us.ibm.com

Frederick R. Reiss
IBM Research - Almaden
San Jose, CA
frreiss@us.ibm.com

- 90% of NER papers in top-TM conferences use ML
- 80% of commercial tools and projects are rule-based
  - Commercial IE tools are often essentially rule editors
- Rule-based: Adaptable, controllable, understandable

Chiticariu et al. (2013). "Rule-Based Information Extraction is Dead! Long Live Rule-Based Information Extraction Systems!". EMNLP.

Learning Rules

- Rules can be learnt from gold standard corpora
- Learn characteristics of the searched entities
  - Context words, suffixes, position in sentence, …
  - That appear frequently around / within positive instances
  - That appear rarely elsewhere
- Rule abstraction is vital
  - Word at this position? Around this position? Word like this?
  - Must be a "was" – "verb-pasttense-1stperson-sg" – "verb-1stperson-sg" – "verb" - "lemma must be <be>"…
- Learning rules requires annotated gold standard corpora
- ML-based NER is about learning rules systematically
  - Next topic

Content of this Lecture

- Named Entity Recognition
  - Dictionary-based approaches
  - Rule-based approaches
  - Machine Learning-based approaches
    - NER as classification
    - Sequential tagging: HMMs, MEMMs, CRFs
- Named Entity Normalization
- Case studies

Classification-Based NER

- General idea: Classify each token as entity or not
  - Learn model based on manually annotated training text
  - Refinement: BIO scheme: "B – first token of an entity" – "I – token within entity" – "O – outside of entity"
  - Being even finer generates trade-off with seeing enough examples in the training data
- Performance depends on feature set
  - Feature engineering: Find properties of tokens that could be characteristic for the search entities
  - Typically one defines a very large feature set and let the classifier decide which ones are decisive (see slides on classification)
- Sequence of tokens can be incorporated by using context features

Typical Features

- [For gene / protein name recognition]
- Surface features
  - Character uni-, bi-, tri-grams
  - POS tag
  - Length in character
  - Has capital letters, all caps, more capital letters than non-cap
  - Has Greek/Roman letters, special characters, digits, all digits
    - 3'-mRNA, 5-alpha-reductase, EST94F88G, …
  - Abstraction: Is of class DDUU, DDSS, DDCDD, …
    - Digits, small case letter, upper case letter, special characters, …
    - Max include contraction: 1.999.000,99 -> D.DDD.DDD,DD -> D.D.D,D
  - …

More Features

- Context features
  - POS tag of surrounding tokens
  - NER tag of preceding tokens (if we only go left-to-right)
  - Presence of indicator words within a certain distance
    - Protein, human, enzyme, plasma, …
- External knowledge
  - Token (or closed-by tokens) matches in a dictionary
- Memory
  - Most frequent tag for this token in texts
  - Most frequent tag for surrounding tokens in corpus
- Others (creativity!)
  - E.g. Number of matches in Google versus PubMed

Classifiers and Ensembles

- Popular choice: SVM / Maximum Entropy
- Ensembles: Use different classifiers and vote
- Example results (more examples later)
  - Different entity types in Spanish; MxE: Max-entropy; TMB: 1-NN neighbor; HMM: Hidden Markov Model

Classification	LOC	MISC	ORG	PER
MxE24$_1$	77.81	57.49	78.83	85.41
TMB24	75.49	53.19	77.44	83.89
MxE25	78.27	58.22	78.64	85.60
TMB25$_2$	75.15	52.94	77.79	85.36
HMM$_3$	71.15	45.69	72.95	70.20
Voting$_{1,2,3}$	78.46	57.00	78.93	86.52

Source: Kozareva, JRC Workshop, 2005

Post-Processing

- Typical problem with multi-token entities: Some tokens are tagged correctly, others not

Gesetz zur effektiven Steuereintreibung

Das 2019 im Bundestag erlassene Gesetz zur effektiven Steuereintreibung wurde von der Opposition…

- Typical solution: Post-Processing based on POS tags
  - If one token of a noun phrase was tagged, tag the entire phrase
  - More conservative: If a noun was tagged, tag all its adjectives
  - Sequential: In case of "B O B" and O has POS tags other than verb or noun or ",": Rewrite O into I, second B into I
  - General: Rewrite "O B B O" into "O B I O" (if nothing in between)
  - Again: Rules can be learned (see syntagmatic POS tagging)

Advantages

- Usually better results than pure dictionary-based NER
  - Providing sufficient and high quality training data
- Problems with multi-token names
- Recognizes unseen entities through abstraction by feature'ization
  - Provided a proper feature set
- "Only" needs an annotated corpus, learning is automatic
  - The larger the better
  - Large corpora are very costly to create
- Reuse of corpora is surprisingly difficult
  - Training data often is surprisingly task-specific
  - Look at cross-corpus results

Disadvantages

- Slower than dictionary-based NER
  - Depending on ML-method
  - This is a killer argument for truly large corpora
- Needs large amount of high-quality training data
  - But high quality NER always requires much manual work, e.g., obtaining high quality dictionaries
- Requires additional NEN step
    - This is a killer argument in practice
- May yield mysterious results that are difficult to tweak
  - Difficult to explain to a user
  - Difficult to tune ("do never tag this word" – black list?)
  - Both are killer arguments in practice

Content of this Lecture

- Named Entity Recognition
  - Dictionary-based approaches
  - Rule-based approaches
  - Machine Learning-based approached
    - NER as classification
    - Sequential tagging: HMMs, MEMMs, CRFs
- Named Entity Normalization
- Case studies

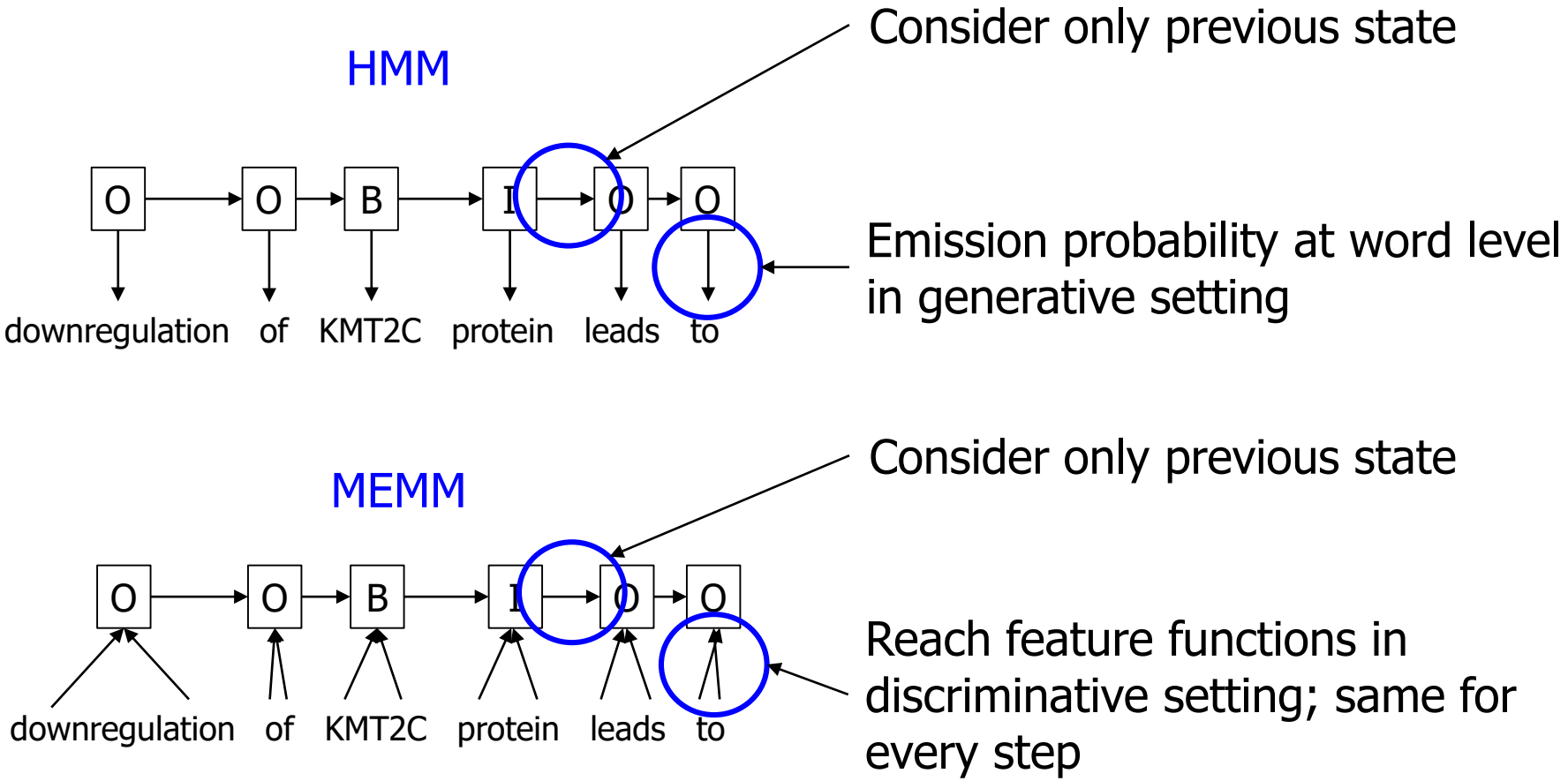
Sequential Tagging using HMMs

- Recall POS tagging with HMMs
  - Fix a set of classes (POS tags)
  - Learn probabilities as state transitions and emissions
  - Encode as Hidden Markov Model
  - Given a new text, find most probable sequence of tags (Viterbi)
- Can readily be applied to NER – with proper tag set
  - Very popular: BIO
- But: Using only tag sequence is not enough for achieving high quality NER
  - Too coarse-grained (only three classes)
  - We need to look at the words and their features, not just their tags

MEMM: Maximum Entropy Markov Models (sketch)

- HMMs are generative models (like Naive Bayes)
- MEMM: A discriminative sequential classifier
  - We predict output (e.g. BIO) from sequential observations (token)
  - MEMM: Transition probabilities are conditional on "observations"
    - Observations are represented by feature functions
    - May encode arbitrary (binary) features
      - "is a noun", "has capital letter", …
  - ME principle to learn conditional transition probabilities is applied separately for each transition from a state q to all next states
    - High-order models are possible
  - Training: GIS algorithm for each state as in ME classification
  - Decoding: Variation of Viterbi algorithm

Visual Explanation



MEMM (and HMM): Label Bias Problem

- MEMM learn a model for each state and its successors
- MEMM thus only learns local models – pairs of states
- But different states have differently many (possible) successors
  - Not much of an issue in NER; but, e.g., real problem in POS
- Inherently, transitions from states with fewer successor states get higher probabilities
  - Because outgoing probabilities must sum to 1 in each state
- These states will dominate inference – appear more often than justified

Conditional Random Fields

- Lafferty, McCallum, Pereira. "Conditional random fields: Probabilistic models for segmenting and labeling sequence data.„. Technical Report, Upenn (2001).
- For long, CRF were state-of-the-art in NER
  - Now probably being replaced by LSTM-CRFs (later)
- CRFs are a mixture of MEMs and HMMs
  - MEM: Represent data by binary feature functions, learn a global model always assessing the entire input/output pair, learn weights of features using iterative optimization methods – but disregard sequence of events
  - HMM: Focus on sequence of states assuming Markov chain property – but mostly disregard observations

Linear Chain CRFs

- We only look at a restricted class: Linear Chain CRF
  - Makes learning and inference more efficient
  - Standard for NER – linear chain of tokens
- Assume a sequence X of token, a sequence Y of labels, and a set f_i of feature functions of the following forms
  - State features: $s(y_i, X, i)$ model an observation at position i of the input X when y_i is the label at position i
    - E.g. s("B", X, i) = 1 iff word at position i contains "ase"; 0 otherwise
  - Transition features: $t(y_{i-1}, y_i, X, i)$ model an observation at position i of the input X when y_i is the label at position i and y_{i-1} is the label at position i-1
    - E.g. s("B", "I", X, i) = 1 iff word at position i-1 has label "B" and word at position i has POS tag "NNP"; 0 otherwise

Model

- A linear-chain CRF computes the conditional probability of the entire tag sequence Y given the entire input sequence X as

$$p(Y|X) = \frac{1}{Z} \exp\left(\sum_{j=1}^{k}\sum_{i=1}^{n} \alpha_j f_j(y_{i-1}, y_i, X, i)\right)$$

- With
  - n: Length of the input sequence, i.e., n=|X|=|Y|
  - k: Number of feature functions
  - Z: Normalization constant
  - α_i: Parameters that must be learned from training data
  - For state features, ignore the parameter y_{i-1}

Power

- A linear-chain CRF computes the cond. probability of the entire tag sequence Y given the entire input sequence X as

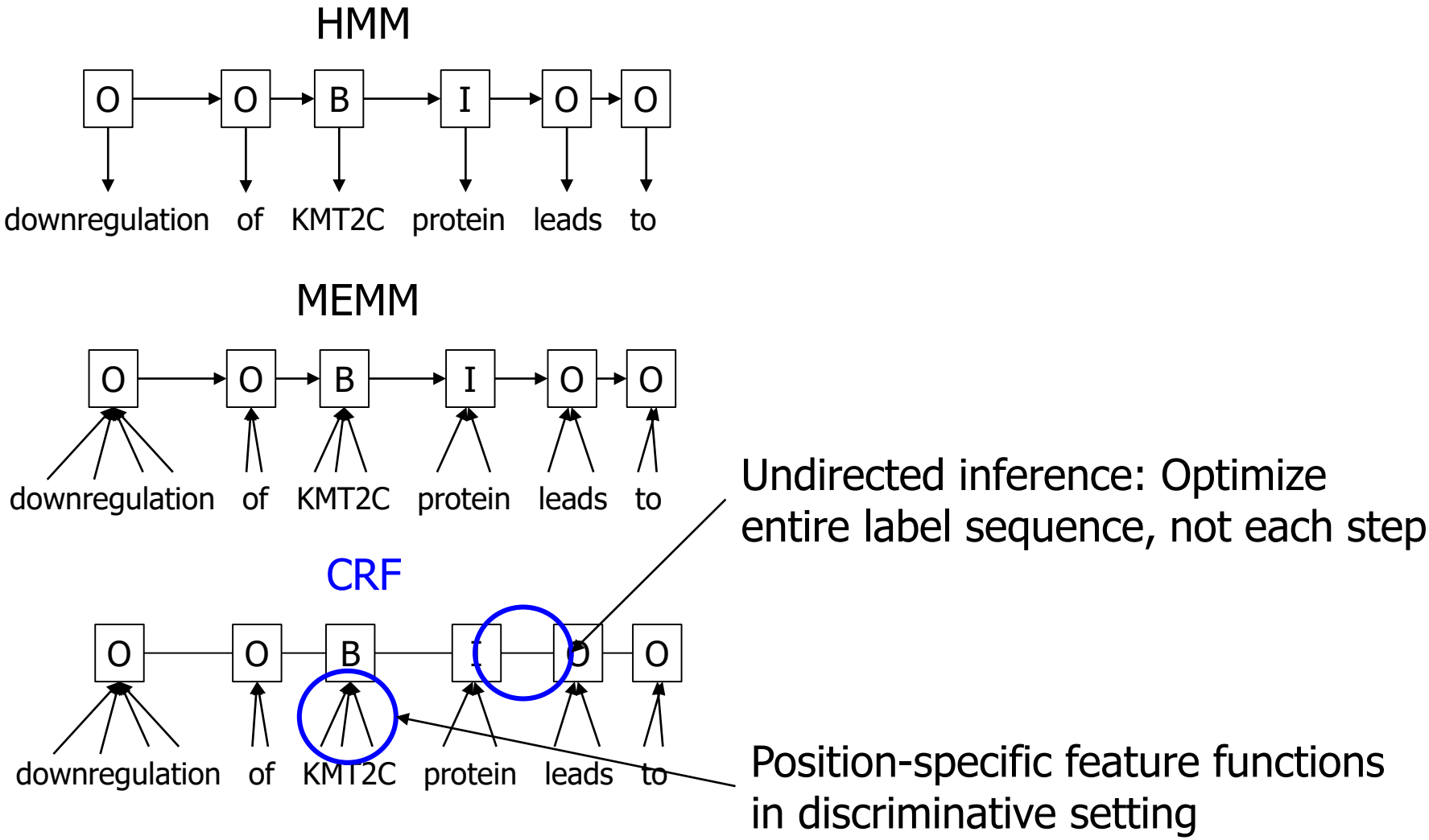
$$p(Y|X) = \frac{1}{Z}\exp\left(\sum_{j=1}^{k}\sum_{i=1}^{n}\alpha_j f_j(y_{i-1}, y_i, X, i)\right)$$

- This is a very powerful model (if trained successfully)
  - We always look at the entire sequence. Correlations between features looking at any position and any property are included
  - Features look at all positions of the input with individual features. Recall that features are defined conditional on position i
  - There is no "direction" – probability of y_i depend on y_{i-1} and y_{i+1}
    - In a global model, we may look into the past and into the future
    - Higher order models are possible – more difficult to train

Algorithms

- Learning parameters α_i: Gradient descent (as for MEM)
  - Optimization problem is convex
- Finding optimal tag sequence
  - For a given Y, computing p(Y|X) is simple – a large sum
  - As for HMMs, we cannot compute p(Y|X) for all possible Y – there are exponentially many
  - Fortunately, dynamic programming still works
    - Any subsequence of an optimal tag sequence is optimal and vice versa
    - Iteratively compute optimal tag sequences for pairs, triples, … of tags
    - Leads to an $O(n^2)$ algorithm similar to Viterbi

Visual Explanation



Comparison

	HMM	MEMM	CRF
Type	Generative	Discriminative	Discriminative
Model	Local	Local	Global
Decoding method	Viterbi-style	Viterbi-style	Viterbi-style
Independence assumption (token-next state)	Yes	No	No
Arbitrary feature functions	No (difficult)	Yes	Yes
Label bias problem	Yes	Yes	No
Learning	Fast	Fast	Slow
Decoding	Fast	Fast	Fast

Content of this Lecture

- Named Entity Recognition
- Named Entity Normalization
- Case studies

Named Entity Normalization (NEN)

- "It is a gene – but which gene?"
- NEN maps each entity to a canonical ID
  - World coordinates of geo-locations
  - RefSeq-IDs of genes
  - Passport / social security numbers of persons
  - ISBN of books
  - Orchid-ID of researchers
  - etc.
- "Canonical" is always domain-specific
  - And often not unique: RefSeq, NCBI gene, ensembl, uniprot, …

NEN and Information Integration

- NEN is a prerequisite to link entities to further information
  - No information integration without NEN
  - NER without NEN has very few (if any) practical applications
- Other names: Entity linking, entity grounding
  - Given a web page – link important parts (entities, key terms) to further information (e.g. Wikipedia, other articles, …)
  - Given an reclamation coming in per mail – link to product ID, customer ID, supplier ID, agent ID, …
- Especially linking to Wikipedia, FreeBase, YAGO etc. is a hot research topic
  - E.g. Google knowledge graph: Recognize entities in search queries and "link" to entities of knowledge graph
  - Links unstructured queries to structured data

Simple NEN Algorithm

- Simple method: Given a mention, find the most similar term in a dictionary of all names of entities of this class
- "Similar" may use the same function as similarity-based dictionary NER
  - Dictionary-based NER has "built-in" NEN
- Difference: We must choose a dictionary entry, no matter how dissimilar the most similar one is
  - Or we return "nil"
- Advantage: Simple, fast

Disadvantages

No resolution of homonyms

Low performance in case of very similar names

Synonyms of EGFR (with ed=1):

Dmel_CG10079, C-erb, CG10079, D-EGFR, D-Egf, DEGFR, DER, DER flb, DER/EGFR, DER/top, DER/torpedo, DER1, DEgfr, Degfr, Der, DmHD-33, Dmel\CG10079, EC2-4, EFG-R, EGF-R, EGFR, EGFr, EGfr, EK2-6, Egf, Egf-r, EgfR, El, Elp, Elp-1, Elp-B1, Elp-B1RB1, Flb, HD-33, TOP, Top, Tor, Torpedo/DER, Torpedo/Egfr, c-erbB, d-egf-r, dEGFR, dEGFR1, dEGFr, dEgfr, degfr, der, egfr, flb, l(2)05351, l(2)09261, l(2)57DEFa, l(2)57EFa, l(2)57Ea, mor1, top, top/DER, top/flb, torpedo/Egfr, torpedo/egfr

Advanced Methods

- Purely syntactic methods quickly reach their limits
- Improving performance requires context
  - Context of entity in text: Surrounding sentence / paragraph
    - Or multiple sentences / paragraph is entity occurs more then once
    - "One-sense-per-discourse" assumption
  - Context of entity in dictionary: Find representative texts
    - E.g. Wikipedia page
    - E.g. Description of gene in Entrez Gene
    - We need such text(s) for all entities in dictionary (!)
    - Often highly varying length and quality – normalization problem
- Two step framework
  - First find all "similar" entities in dictionary ("candidates")
  - Resolve concrete entity using context information ("reranking")

Matching Context

- Define a context similarity function and chose entity with most similar context
  - E.g. cosine of bow-representation of context texts
- Treat as multi-class classification problem: One class per entity
  - Difficult with many entities and high variance in length and quality of context texts

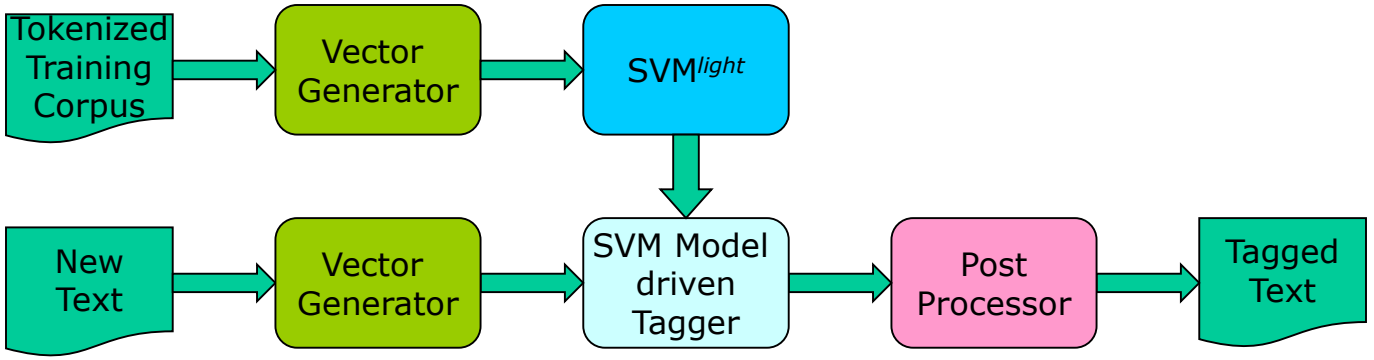
Content of this Lecture

- Named Entity Recognition
- Named Entity Normalization
- Case studies
  - BioCreative
  - MUC conferences
  - Predicting ICD-10 codes

BioCreative Cup 2004

- Critical Assessment of Information Extraction Systems in Biology
- International competition, three tasks
- Training data and evaluation script provided by organizers in cooperation with database curators (Swiss-Prot)
- Test data available for one week
- Evaluation of all submissions by (published) scripts
- Major boost: Top systems reached 84 F1-measure
  - Previous best systems around 60 F1-Measure
  - Possibly not much further improvements since then
  - Fields splits up: Species, NER/NEN, NER/PPI, …

Example: SVM for NER



- Corpus of 7500 sentences
  - 140.000 non-gene words
- SVM^{light} on different feature sets
- Dictionary compiled from Genbank, HUGO, MGD, YDB
- Post-processing for compound gene names

Features

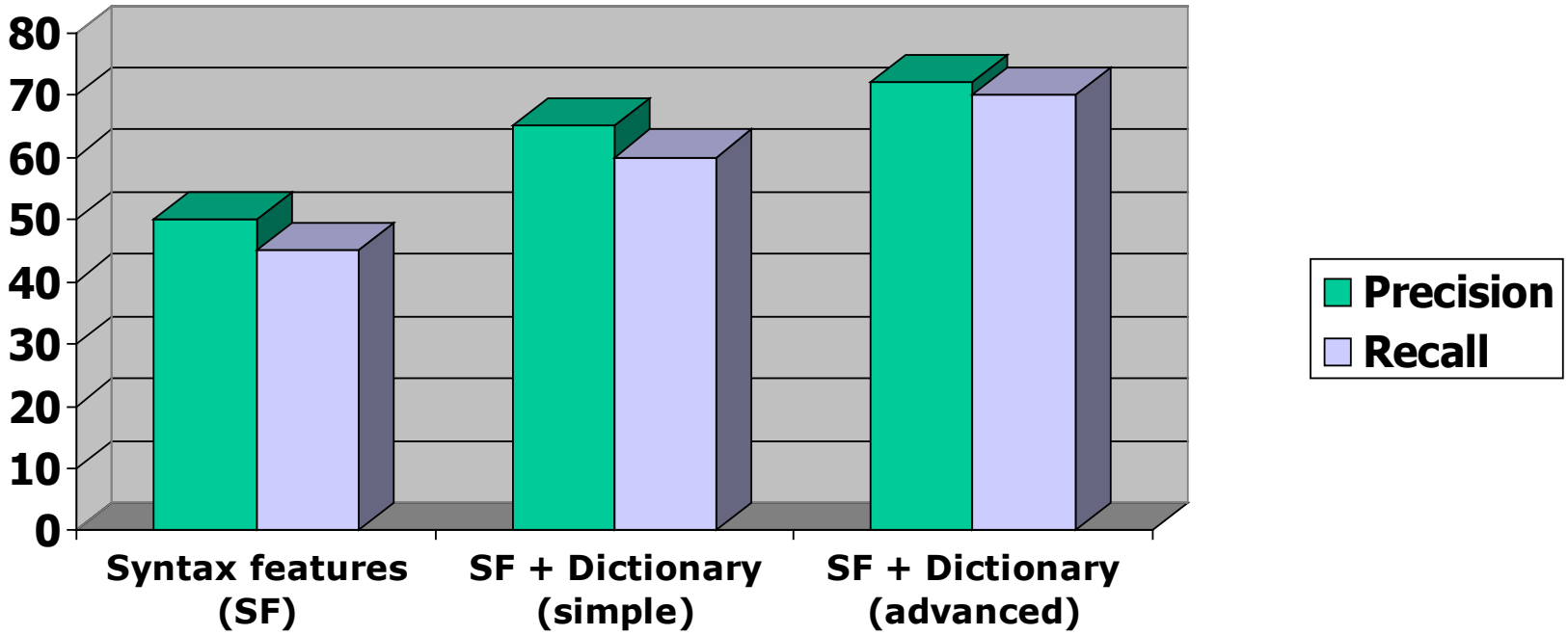
Feature	Weight	Example
Word	tf * idf	kinase
n-grams		
N=1	tf * idf	k, i, n, a, s, e
N=2	tf * idf	ki, in, na, as, se
N=3	tf * idf	kin, ina, nas, ase
Special signs		
HasNumbers	[1|0]	p300
HasCapitals	[1|0]	abLIM
AllCaps	[1|0]	DMD
InitCap	[1|0]	Pax
HasNumbers & Letters	[1|0]	cMOAT2, EST90757
Context		
predecessing word	[1|0]	Gene
succeeding word	[1|0]	Product
distance to keywords	1/(1+dist)	(list of 15)
Dictionary		
Word match	[1|0]	
Phrase match	[1|0]	

Post-processing

- SVM detects only single token candidates
- Most gene names are multi-token names
- Expand detected single-token genes based on set of heuristic rules (found in an unsystematic manner)

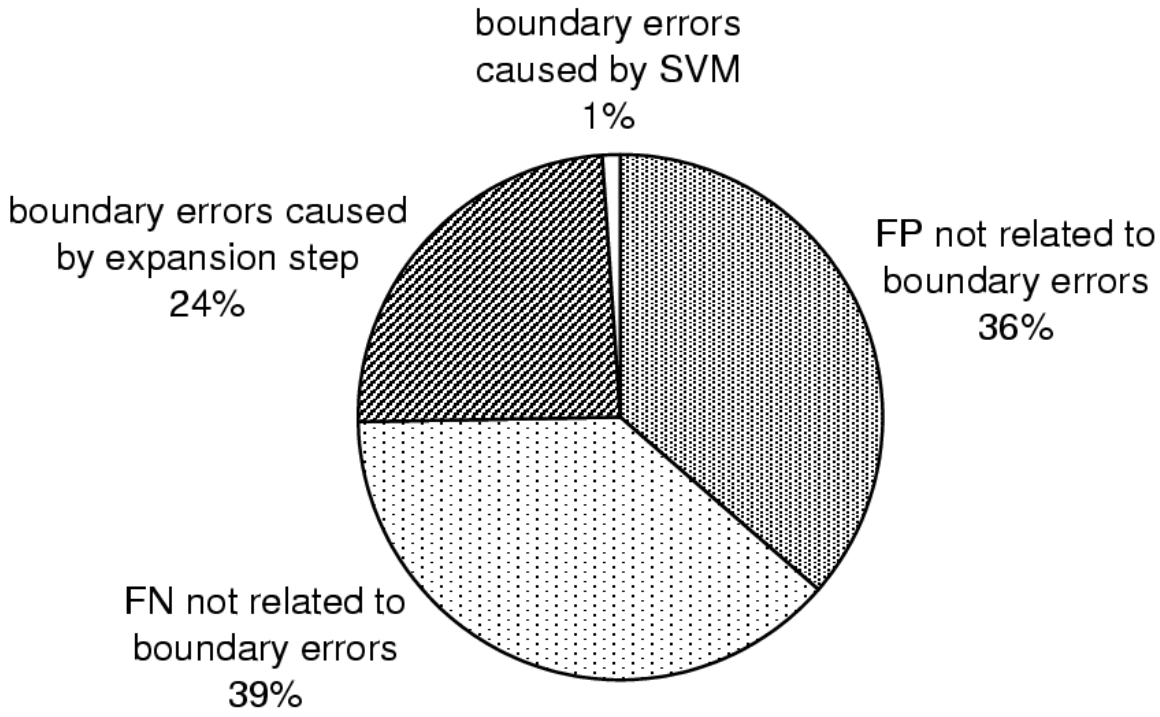
GENE NN*	➔	GENE GENE
NN* GENE	➔	GENE GENE
GENE (NN)	➔	GENE (GENE)
GENE protein	➔	GENE GENE
GENE ADJ GENE	➔	GENE GENE GENE

Performance



- Best result for BioCreative Cup: 73 F-measure
  - 12 percentage point increase by post-processing only
- Raises from 73 to 83 for loose evaluation

Where did we Fail?



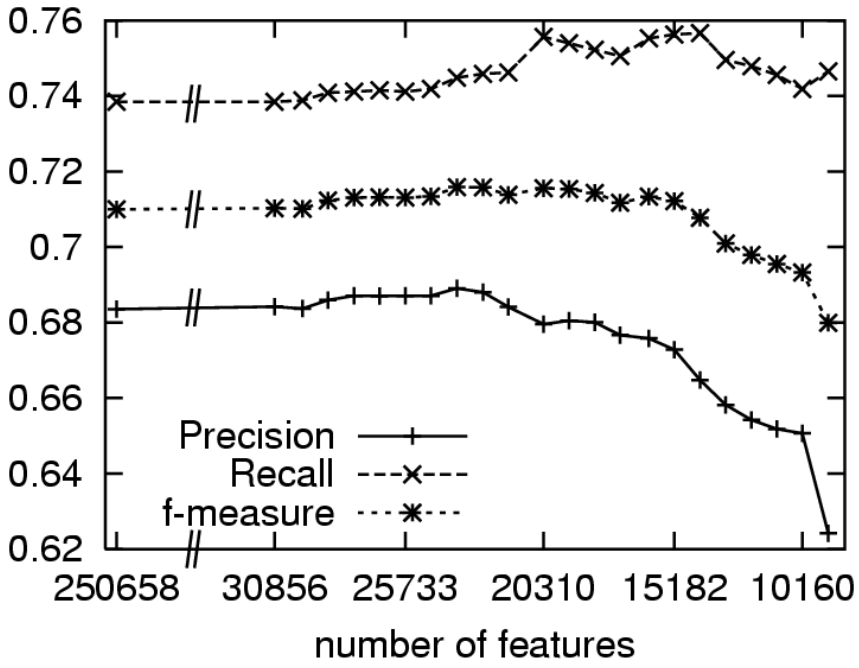
- „Boundary error" – problems with multi-word phrases
- >70% of errors are token classification errors from SVM

Impact of Feature Classes

Feature	Example	Short name	Impact		
Token *	`Sro7`	Token	=54%		- *baseline* -
Unseen token *		UToken			
n-grams of token *		1G, 2G,	+15%	1..4	grams, P+, R++
			+14%		1..3-grams
Previous & next tokens		P/NToke	-5%	[1,1	window, P+, R-
			-6%		2,2]-window
n-grams of tokens in window		2PG/2NG..			
Prefixes, suffixes		1P,2P,3P,1..	±0		
Stop word	`the, or`	Stop	-5%	10.0	0 words, P+, R-
			-1%	1.0	0 words, P+, R-
			-.5%	10	words, P+, R-
POS ta					
Initial					
All cha					
Upper					
Upper					
Single					
Two ca					
Capital					
Lower					
Special					
Charac					
Numbe					
Letters					
Digit,					
Greek					
Roman					
Number followed by '%' ○	`75.0 %`	percentag	-.1%		P-, R-
DNA, RNA sequences ○	`ACCGT`	DNA, RN	-.1%		P-, R-
Longest consonant chain *	`Sro7`→ 2	LCC	-2%		P-, R-
Keyword distance *		keyDist	-20%		P+, R-
Gazetteer *		Gaz	-3%		P-, R-
Prev./next token is NEWGENE		PTG, NT	-18%	pr	. only, P+, R-
Tokens + letter surface clues			+2%		P+, R-
Tokens + 1,2,3-grams + greek + roman + letter surface clues			+14%		P+, R++
Tokens + 1,2,3-grams + keyDist + Gaz + LCC + special + combi + allCaps + initCap *			+ 16%		P+, R++
Tokens + 1,2,3,4-grams + keyDist + Gaz + LCC + special + combi + allCaps + initCap + lowMix ○			+ 18%		P+, R++

Tokens + letter surface clues	+2%	P+, R-
Tokens + 1,2,3-grams + greek + roman + letter surface clues	+14%	P+, R++
Tokens + 1,2,3-grams + keyDist + Gaz + LCC + special + combi + allCaps + initCap *	+ 16%	P+, R++
Tokens + 1,2,3,4-grams + keyDist + Gaz + LCC + special + combi + allCaps + initCap + lowMix ○	+ 18%	P+, R++

Do we need them all?



- Repeated elimination of 5% least discriminating features
- Eliminating 95% of features costs only 2% F-Measure

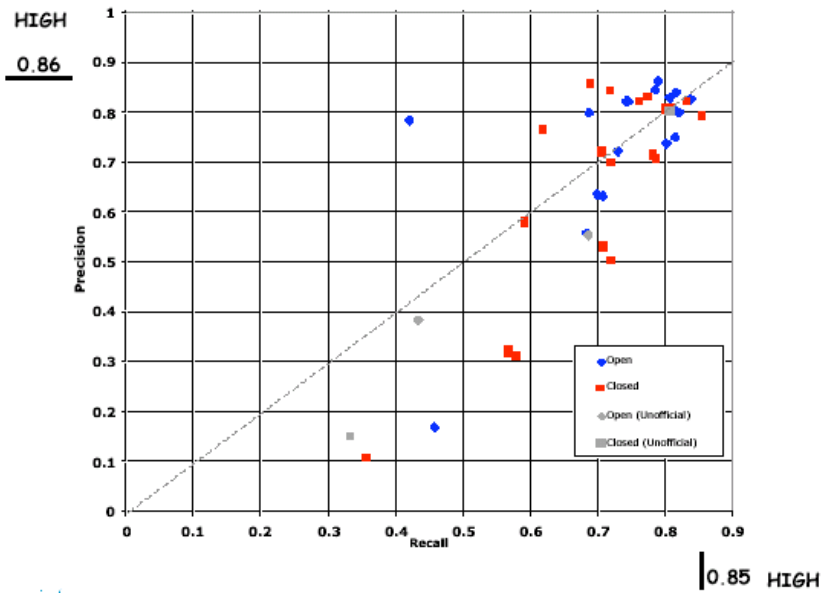
Which Ones?

- Single features from different classes are among the most important ones
- Difficult to remove entire classes of features

Feature	Class	Weight
	Gaz	1.497386
insulin	Token	0.632708
protein	Token	0.628168
kinase	Token	0.608392
human	Token	0.536695
proteins	Token	0.535368
	greek	0.498111
	combi	0.489201
serum	Token	0.480326
	lowerUpper	0.457806
	singleCap	0.438028
factor	Token	0.438028
wild-type	Token	0.389359
	initCaps	0.366269
mutants	Token	0.340689
genes	Token	0.340352
promoter	Token	0.327395
receptor	Token	0.323412
polymerase	Token	0.305972
complex	Token	0.292019
receptors	Token	0.292019
c-myc	Token	0.292019
sites	Token	0.243349
mutant	Token	0.243349
domain	Token	0.231541
sequence	Token	0.216691
sequences	Token	0.216683
domains	Token	0.215116
	specialnumber	0.205077
isoforms	Token	0.194679
	specialupperCase	0.179926
	capMixLetters	0.179394

Other Systems [BioCreative 2004]

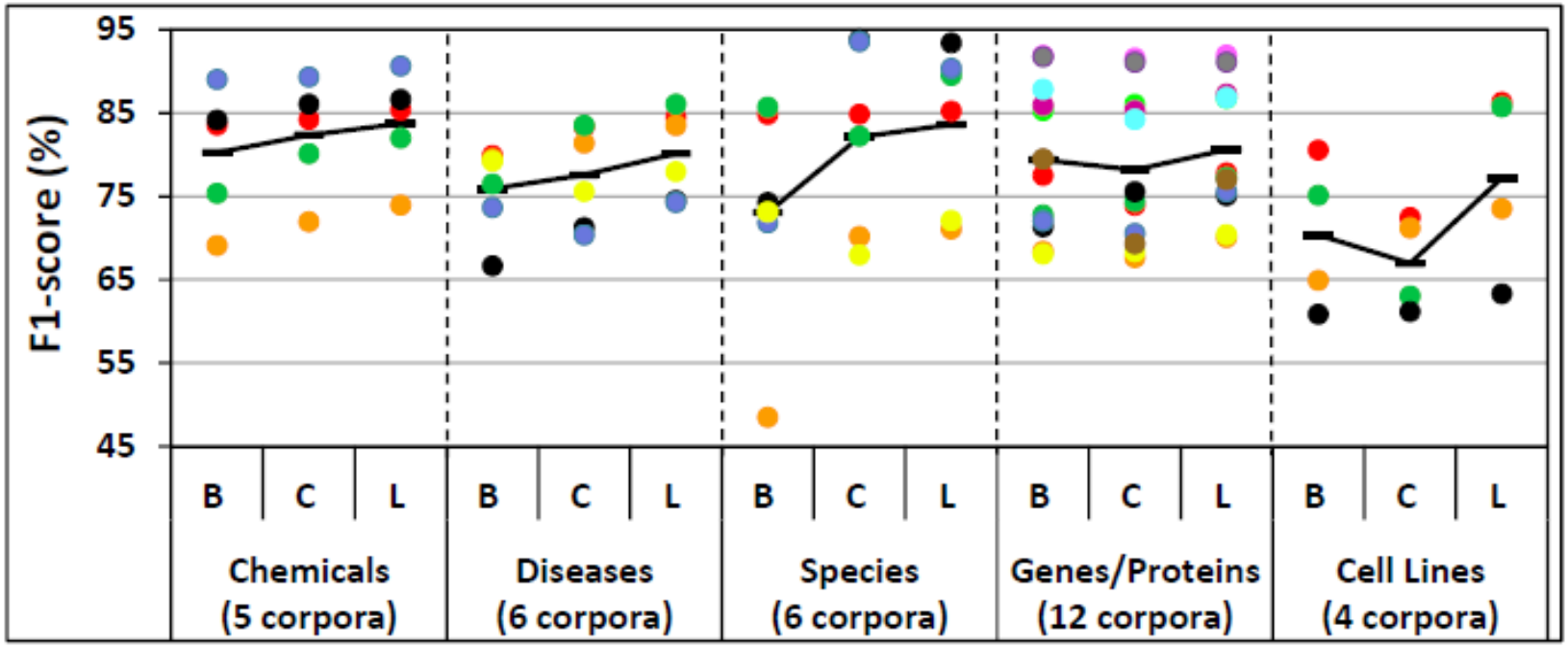
- Best: MMEM or CRF
- Much larger feature sets
- Use of ensembles trained on different corpora
- Current state-of-the-art
  - F-measure ~85%
  - Strongly dependent on eval corpus and entity type
  - Often close to Inter-annotator agreement
  - Loose evaluation reaches +10-20% F1
  - Biomedical entities are much more difficult than persons, companies, etc.



Gene-NER: Why is it hard?

- „Scientists would rather share each other‘s underwear than use each other‘s nomenclature“ [Keith Yamamoto]
- Ambiguous gene names and high number of acronyms
  - The, white, ACL, …
- Small training and eval corpora, mostly only abstracts
- Strict vs. loose matching (up to 20% in F1 difference)
- Generally little agreement on gene names (low IAA)
- Cross-corpus performance
  - All corpora differ in scope
  - Method trained on corpus A performs bad on corpus B
  - Domain Adaptation Problem

State-of-the-Art: LSTM-CRF with Word Embeddings



Content of this Lecture

- Named Entity Recognition
- Dictionary-based approaches
- Rule-based approaches
- ML-based approached
- Case studies
  - BioCreative
  - MUC conferences
  - Predicting ICD-10 codes

Message Understanding Conferences (MUC)

- Large conferences and competitions (1987 – 1998)
- Initiated and funded by DARPA (among other)
- Similar to TREC, but focusing on information extraction / named entity recognition
- Tasks including co-reference resolution
- Template filling / "model-based" IE

Mr. John Smith was appointed CEO of ACME last December 31.

Name:	John Smith
Post:	CEO
Company:	ACME
Date:	December 31

Corpora

Year	Conference	Domain
1987	MUC-I	Navy messages
1989	MUC-II	Navy messages
1991	MUC-3	News about terrorist attacks
1992	MUC-4	News about terrorist attacks
1993	MUC-5	Company news (joint-ventures, micro-electronics production)
1995	MUC-6	Company news (management succession)
1998	MUC-7	Airline company orders

Source: Boullosa, NER

Results (MUC-7, 1998)

Task	Recall (%)	Precision (%)
Named Entity (NE)	92	95
Coreference	63	72
Scenario Template (complete events)	47	70

Systems (MUC-7, 1998)

F-Measure	Error	Recall	Precision
93.39	11	92	95
91.60	14	90	93
90.44	15	89	92
88.80	18	85	93
86.37	22	85	87
85.83	22	83	89
85.31	23	85	86
84.05	26	77	92
83.70	26	79	89
82.61	29	74	93
81.91	28	78	87
77.74	33	76	80
76.43	34	75	78
69.67	44	66	73

Annotators:

97.60	4	98	98
96.95	5	96	98

- Best system is a hybrid between an extensive set of rules and a ME classifier

Context Rule	Assign	Example
Xxxx+ is a? JJ* PROF	PERS	Yuri Gromov is a former director
PERSON-NAME is a? JJ* REL	PERS	John White is beloved brother
Xxxx+, a JJ* PROF,	PERS	White, a retired director,
Xxxx+ ,? whose REL	PERS	Nunberg, whose stepfather
Xxxx+ himself	PERS	White himself
Xxxx+, DD+,	PERS	White, 33,
shares of Xxxx+	ORG	shares of Eagle
PROF of/at/with Xxxx+	ORG	director of Trinity Motors
in/at LOC	LOC	in Washington
Xxxx+ area	LOC	Beribidjan area

Source: Mikheev, Grover, Moens, „DESCRIPTION OF THE LTG SYSTEM USED FOR MUC-7"

Content of this Lecture

- Named Entity Recognition
- Dictionary-based approaches
- Rule-based approaches
- ML-based approached
- Case studies
  - BioCreative
  - MUC conferences
  - Predicting ICD-10 codes (recall from intro)

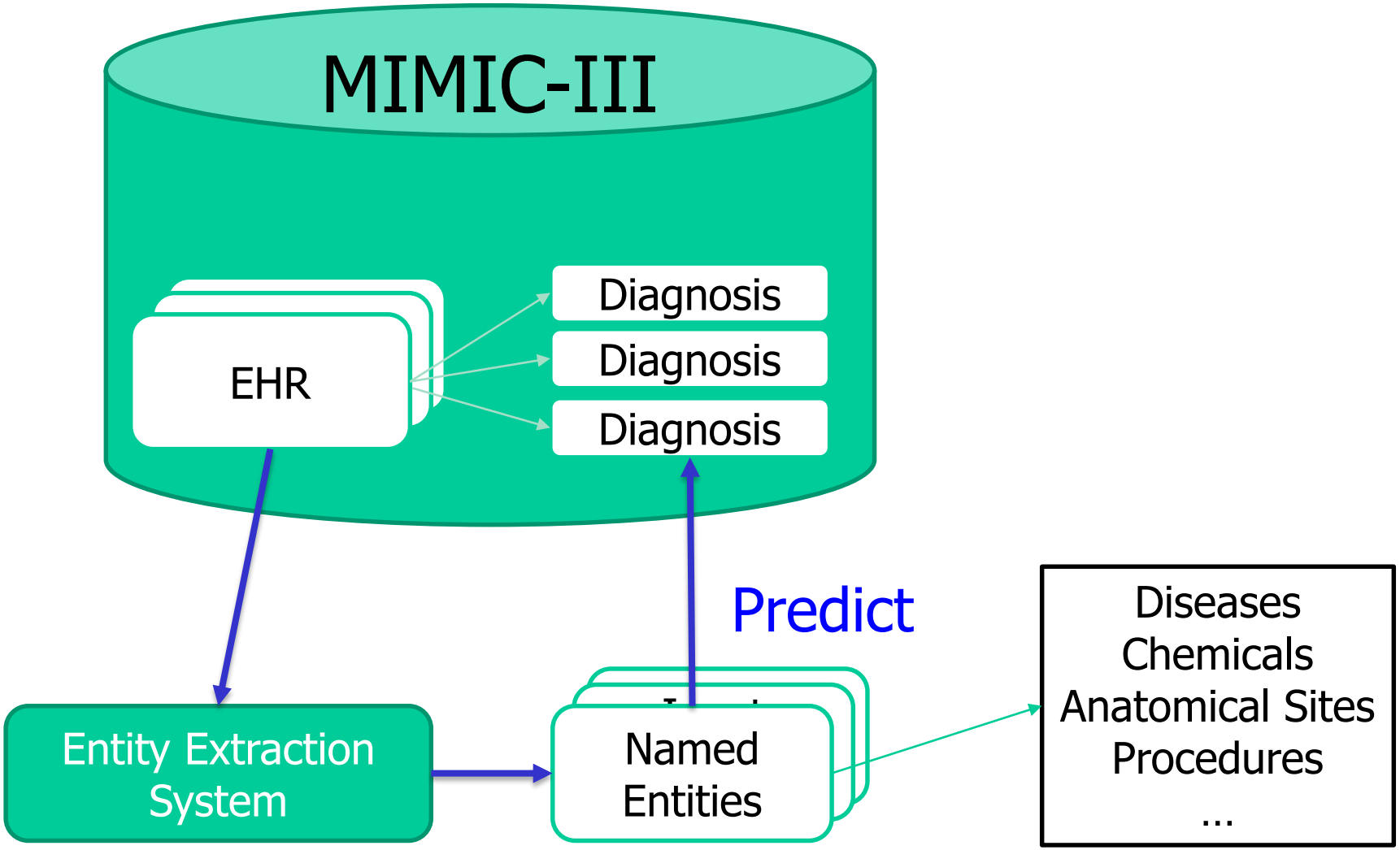
Predicting Disease Codes based on Patient Records

- Medical diagnosis are encoded in fixed vocabularies
  - For accounting, for statistics, for integration, for data mining
- Most important taxonomy: ICD-9/10
  - International Classification of Diseases
  - "codes for diseases, signs and symptoms, abnormal findings, complaints, social circumstances, and external causes of injury or diseases"
  - Roughly 15.000 codes in hierarchical organization
  - DRG: German "disease related groups", derived from ICD-9, used for accounting of medical treatments

Problem

- Proper ICD-10 annotation is vital for any hospital
  - DRG codes: Disease related groups
- Physicians do not use ICD codes for documentation
  - Too clumsy, too many, not precise enough, much relevant information not expressible (temporal development, dosage, …)
- Currently, a "Medizinischer Dokumentarist" reads EHR's and adds DRG codes
- Task: Can we automatically predict ICD codes based on medical records?
  - Results here: J. Bräuer, Clinical Entity Recognition for ICD-9 Code Prediction in Clinical Discharge Summaries, Diplomarbeit, 2017

Architecture

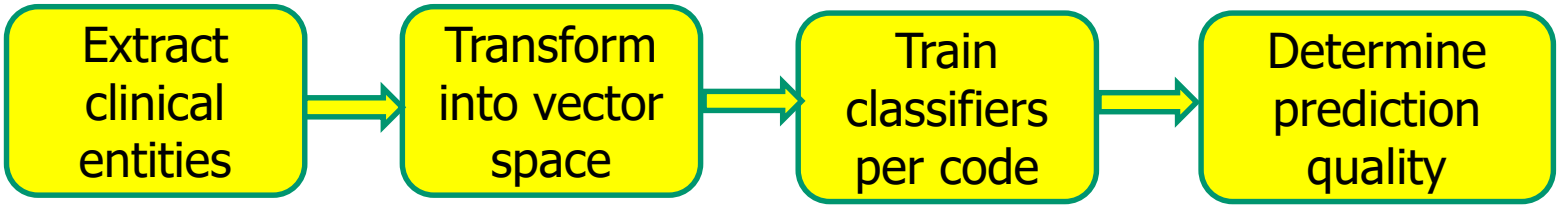


- **DATE OF ADMISSION:** MM/DD/YYYY
- **DATE OF DISCHARGE:** MM/DD/YYYY
- **DISCHARGE DIAGNOSES:**
- 1. Vasovagal syncope, status post fall.
  2. Traumatic arthritis, right knee.
  3. Hypertension.
  6. History of chronic obstructive pulmonary disease.
- **BRIEF HISTORY:** The patient is an (XX)-year-old female with history of previous stroke; hypertension; COPD, stable; renal carcinoma; presenting after a fall and possible syncope. While walking, she accidentally fell to her knees and did hit her head on the ground, near her left eye. Her fall was not observed, but the patient does not profess any loss of consciousness, recalling the entire event. The patient does have a history of previous falls, one of which resulted in a hip fracture. She has had physical therapy and recovered completely from that…
- **DIAGNOSTIC STUDIES:** All x-rays including left foot, right knee, left shoulder and cervical spine showed no acute fractures. The left shoulder did show old healed left humeral head and neck fracture with baseline anterior dislocation. …
- **HOSPITAL COURSE:**
- 1. Fall: The patient was admitted and ruled out for syncopal episode. Echocardiogram was normal, and when the patient was able, …
- 2. Status post fall with trauma: The patient was unable to walk normally secondary to traumatic injury of her knee, causing significant pain and swelling. Although a scan showed no acute fractures, …

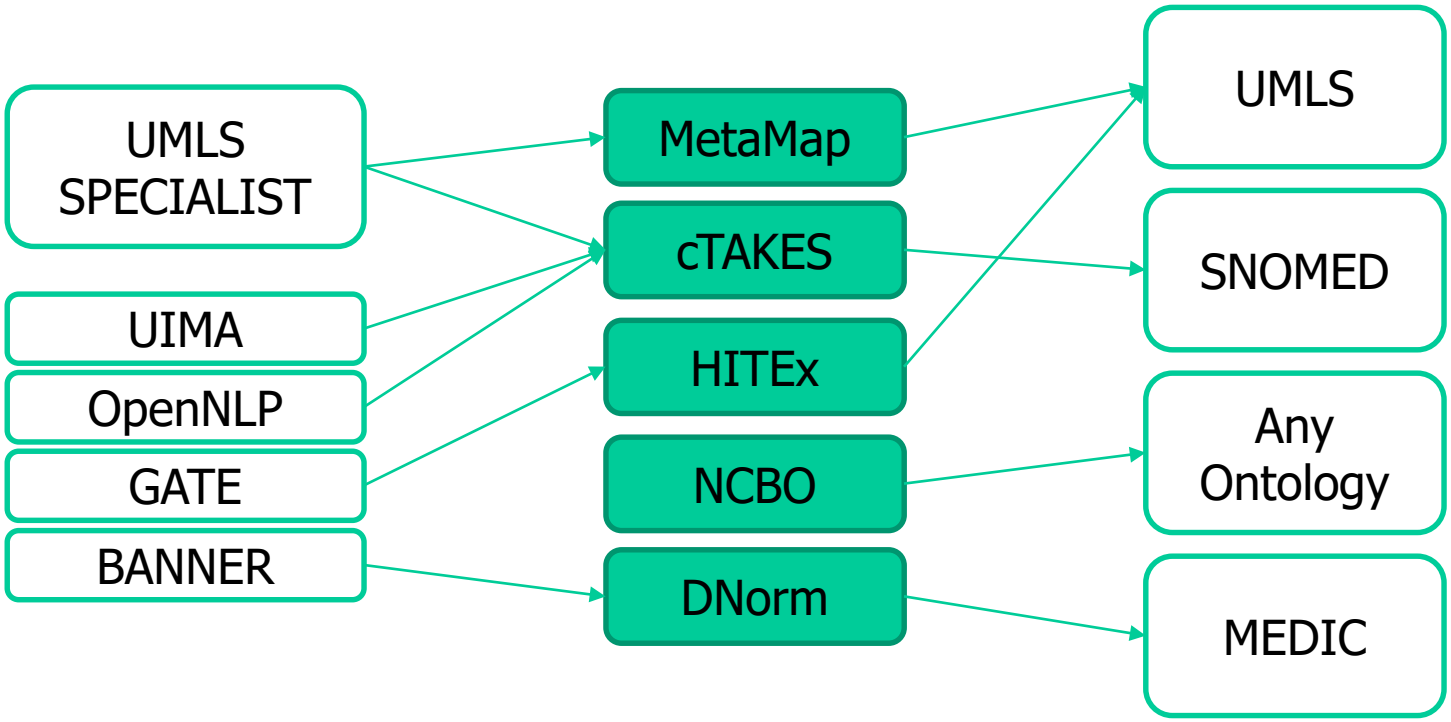
Goals and Methods

- Predict discharge diagnosis based on clinical texts
- Approach 1: Recognize diseases in text (NER-based approach)

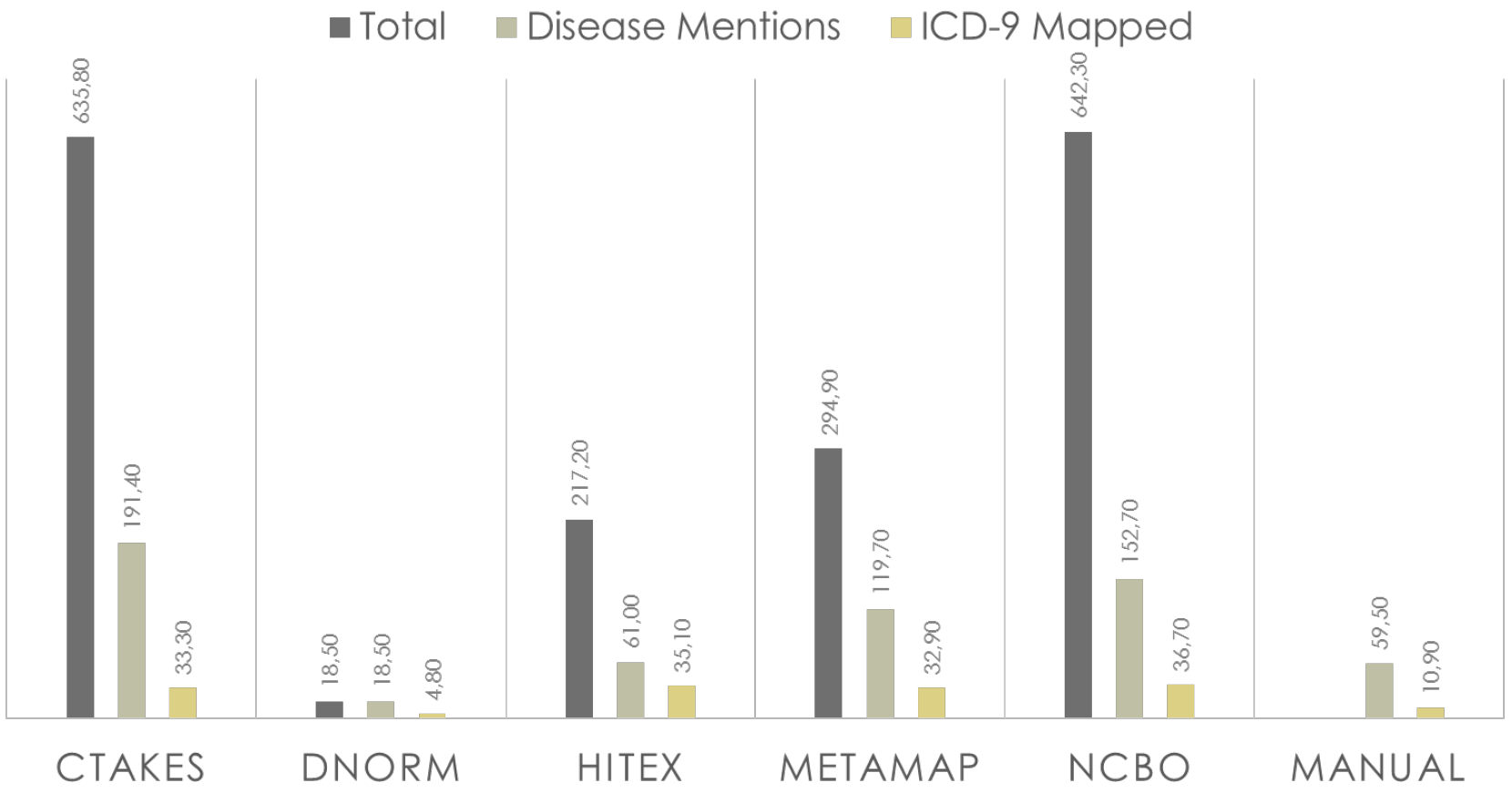
- Approach 2: Predict disease based on (entire, partial) text (classification-based approach)



Medical NER Tools Evaluated

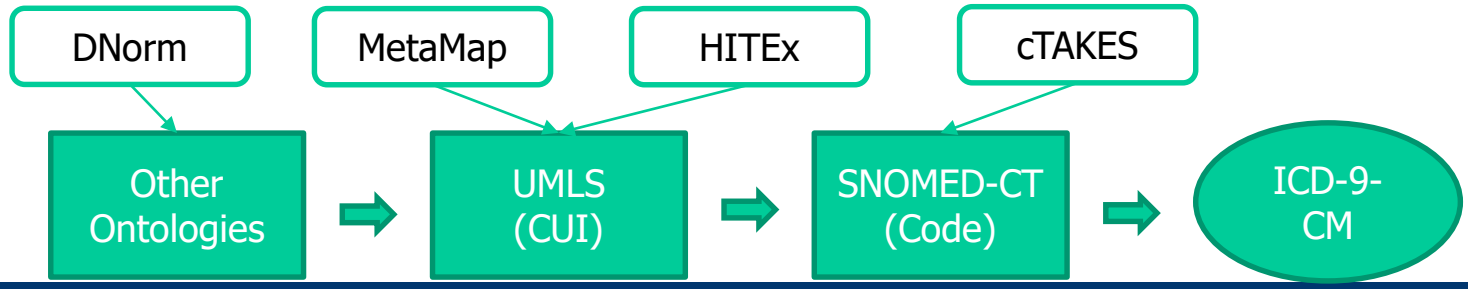


Number of Extracted Concepts (Per Document)



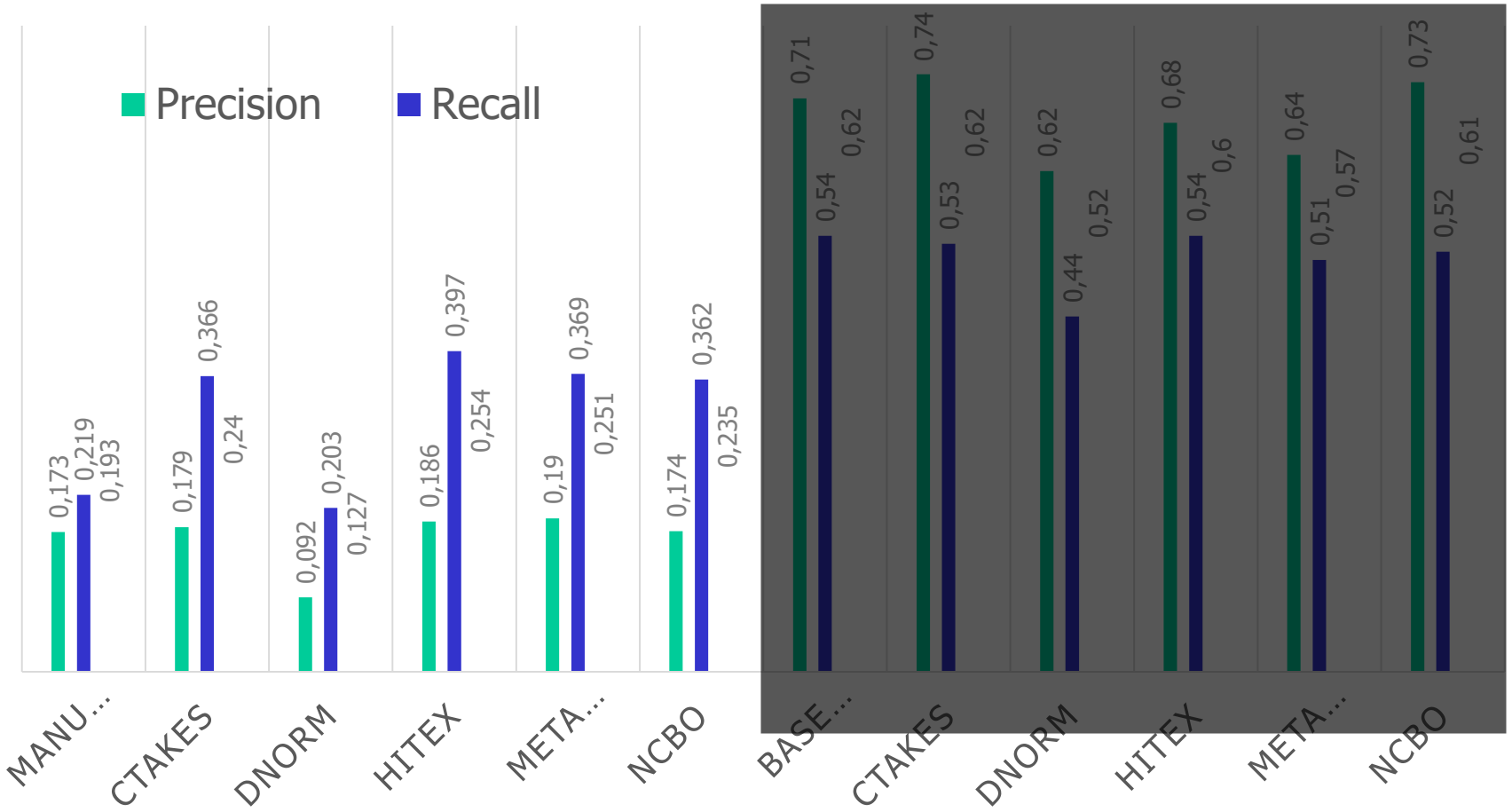
Issues (Typical)

- Hierarchical classification – which level of ICD-9?
  - Higher levels: More training data, few classes, high accuracy
    But: Little value
  - Lower levels: Little training data, many classes, low accuracy
    But: High value
- Mapping between ontologies
  - Concepts with different syntax & synonyms
  - Concepts at different granularities
  - Conflicting subsumption relationships
  - Diverging coverage
  - …



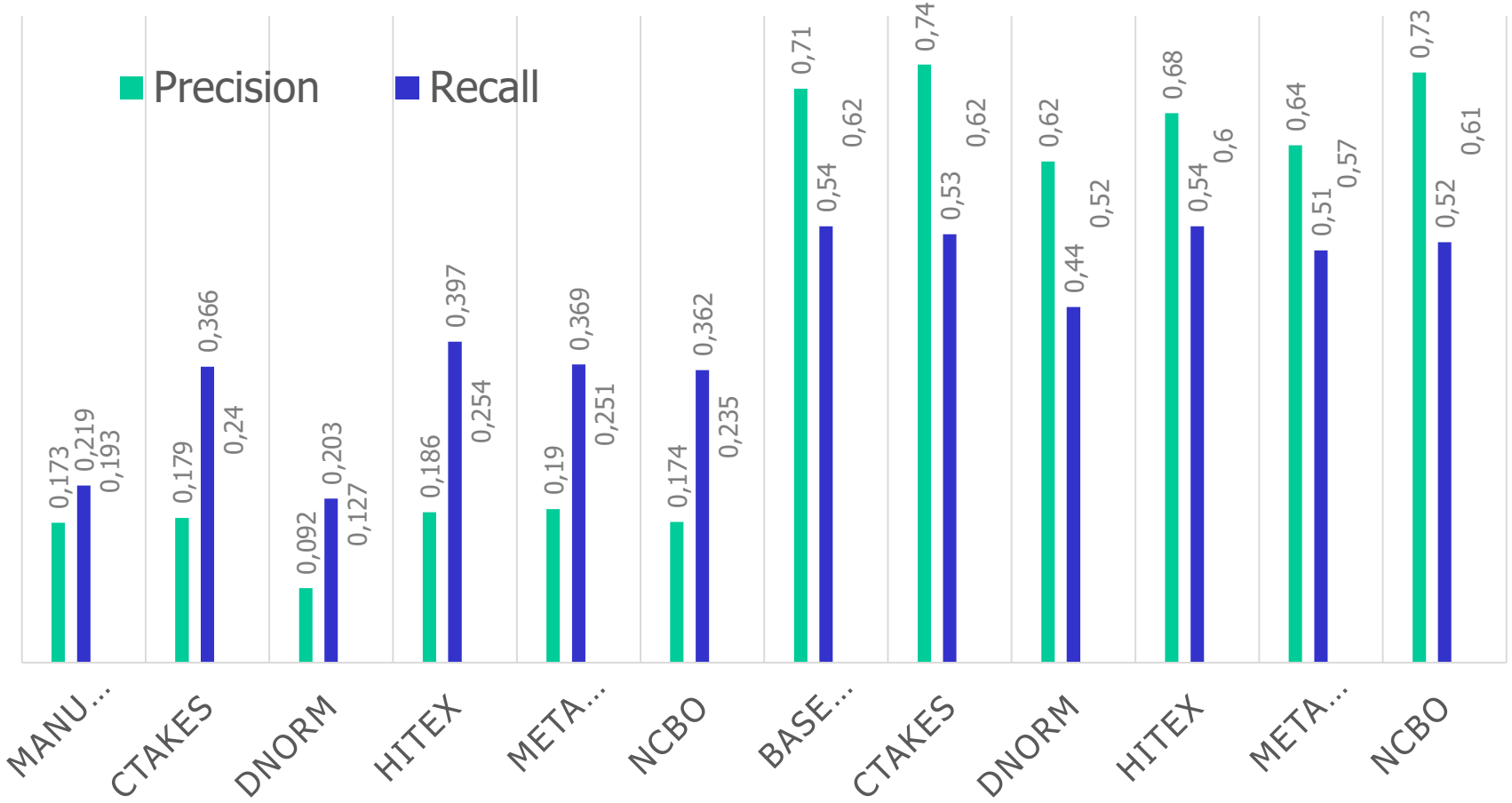
Results / Evaluation

- 50 k discharge summaries
- 7 k classes (diagnosis codes)



Results / Evaluation

- Baseline: 10 k top concepts 7 k
- Train/test split 90% / 10%



Selbsttest

- What is the difference between NER and NEN?
- Describe some syntactic similarity functions for entity names. What is their computational complexity?
- What special problems occur with multi-token entities?
- What is the relationship between a HMM and a CRF? Is any of them strictly more expressive than the other? Why?
- What could be typical surface features company names in Germany? For names of cities and villages?
- How can two contexts for NEN be obtained and compared?
- Describe NEN with textual context as a information retrieval problem. Could Lucene help?