Resource-Lean Lexicon Induction for German Dialects

Robert Litschko, Barbara Plank, Diego Frassinelli

robert.litschko@lmu.de

LREC 2026 Palma

LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

CIS

nlp

mcml Munich Center for Machine Learning

Overview

- Introduction
- Method
- Results
- Conclusion

Introduction

- Performance of NLP tools and LLMs depend on how well language is represented in training data.
- High-resource languages benefit, e.g., from robust translation and retrieval systems.
- Performance degrades under dialect spelling variation.

German ↔ Bavarian

Bürgermeister (mayor) ↔ **Birgermoaster** **Bürgermaischter** **Birgermóaster** **Burgermoaster** **Byrgermoaster** **Bürgermoaster** **Birgermeister** **Birgermaster**

➔ Goal: Build variation dictionaries to bridge dialect gap.

Word-Pair Classification

- Task: Match German lemmas with lexically similar Bavarian words (word pair classification).
- Prior work: LLMs perform poorly at identifying correct translations.[1]

[1] *Make Every Letter Count: Building Dialect Variation Dictionaries from Monolingual Corpora (Litschko et al., 2025)*

Word-Pair Classification

- Task: Match German lemmas with lexically similar Bavarian words (word pair classification).
- Prior work: LLMs perform poorly at identifying correct translations.[1]
- Contribution: Evaluate a statistical model based on string similarity features.

[1] *Make Every Letter Count: Building Dialect Variation Dictionaries from Monolingual Corpora (Litschko et al., 2025)*

Dialect Variation Dictionaries

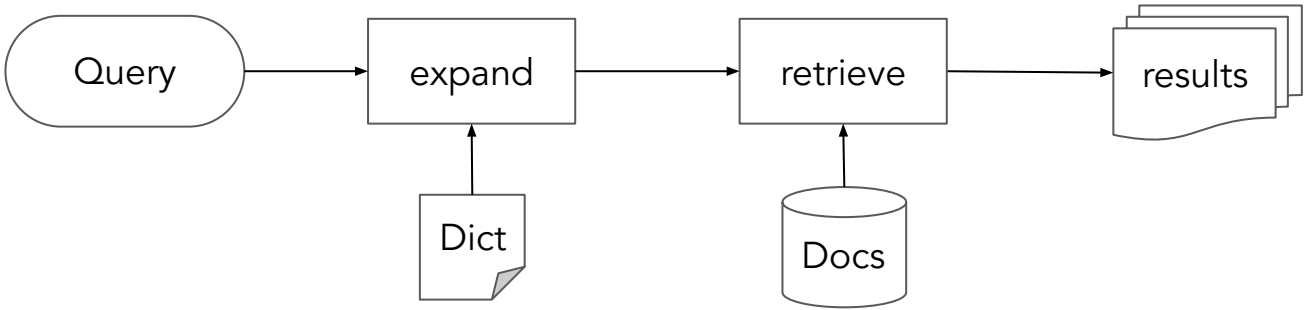
- We use random forests to build dialect variation dictionaries.
- DIALEMMA annotation framework:[1] Induce dictionaries from 100K German lemmas, each paired with ten lexical nearest neighbors.

Dialect	Lemmas	Variants	V/L
als	38.1K	88.1K	2.31
bar	27.6K	51.4K	1.86
ksh	6.9K	9.4K	1.36
pfl	9.1K	13K	1.43
nds	22K	39.5K	1.80

[1]*Make Every Letter Count: Building Dialect Variation Dictionaries from Monolingual Corpora (Litschko et al., 2025)*

Cross-Dialect Information Retrieval

- Task: Given German query, retrieve dialect documents containing keyword spelling variations.[1]
- Dialects: Ripuarian (ksh), Low German (nds), Alemannic (als), Rhine Franconian (pfl), Bavarian (bar).
- Query expansion: add spelling variations from (automatically induced) dictionaries.



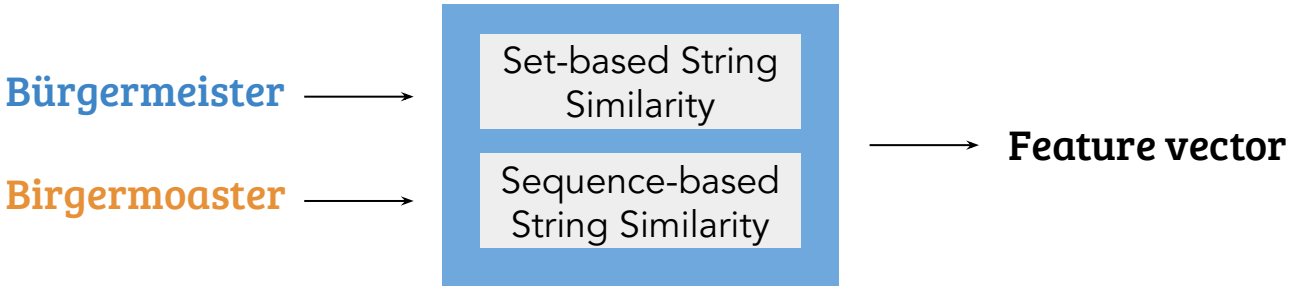
[1] *Cross-Dialect Information Retrieval: Information Access in Low-Resource and High-Variance Languages (Litschko et al., 2025)*

Overview

- Introduction
- Method
- Results
- Conclusion

String Similarity Features

- Our string similarity features are adopted from (Inkpen et al., 2005) and were originally developed for classifying words as cognates or false friends.[1]
- Features are generated from pairs of German lemmas and dialect candidate words.



[1] *Automatic identification of cognates and false friends in French and English (Inkpen et al., 2005)*

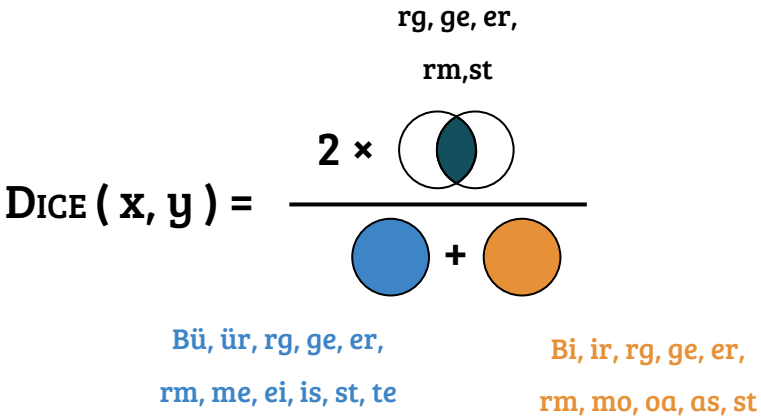
Set-based String Similarity

- DICE coefficient between ngrams(x) and ngrams(y).

$$\text{DICE}(x, y) = \frac{2 \times \left| \text{ngrams}(x) \cap \text{ngrams}(y) \right|}{\left| \text{ngrams}(x) \right| + \left| \text{ngrams}(y) \right|}$$

Set-based String Similarity

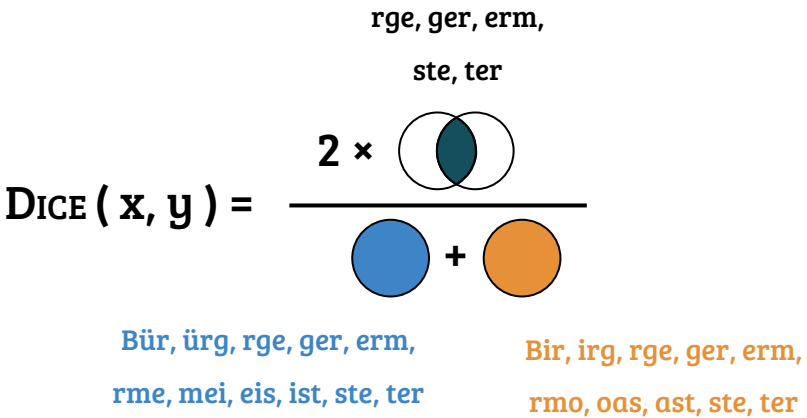
- DICE coefficient between ngrams(x) and ngrams(y).
  - Shared character bigrams[1]



[1] *The use of an association measure based on character structure to identify semantically related pairs of words and document titles (Adamson and Boreham, 1974)*

Set-based String Similarity

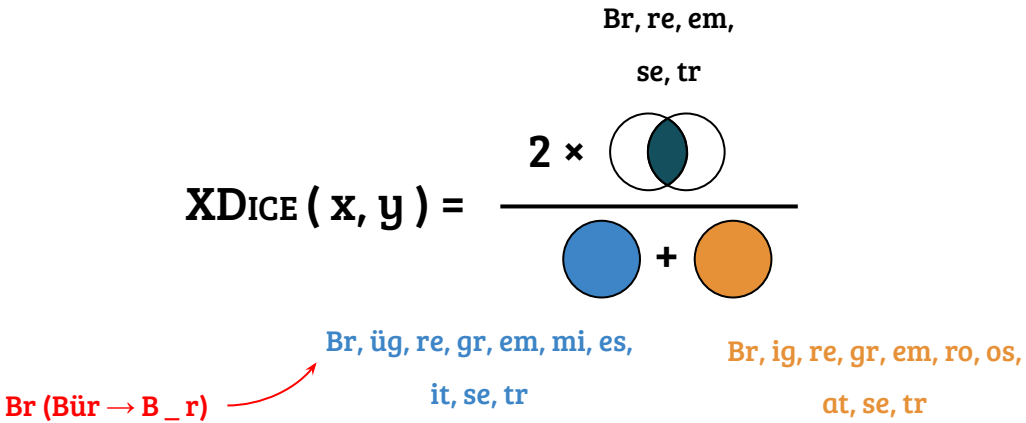
- DICE coefficient between ngrams(x) and ngrams(y).
  - Shared character bigrams, trigrams[1]



[1] *The use of an association measure based on character structure to identify semantically related pairs of words and document titles (Adamson and Boreham, 1974)*

Set-based String Similarity

- DICE coefficient between ngrams(x) and ngrams(y).
  - Shared character bigrams, trigrams, and "extended trigrams" (XDICE).[1]

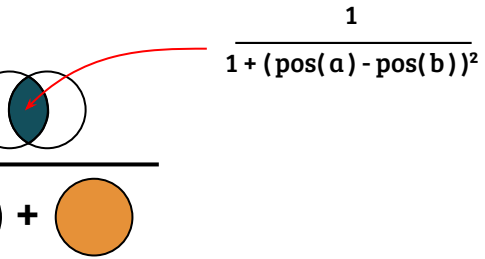


[1] *Word-pair extraction for lexicography (Brew et al., 1996)*

Set-based String Similarity

- DICE coefficient between ngrams(x) and ngrams(y).
  - Shared character bigrams, trigrams, and "extended trigrams" (XDICE).
- XXDICE incorporates positional information.[1]

$$\text{XXDice}(x, y) = \frac{2 \times \sum_{a \in x} \sum_{b \in y} \frac{1}{1 + (\text{pos}(a) - \text{pos}(b))^2}}{|x| + |y|}$$



[1] *Word-pair extraction for lexicography (Brew et al., 1996)*

Sequence-based String Similarity

- Length of longest common prefix (PREFIX)

PREFIX (x, y) = | B | = 1

B ü r g e r m e i s t e r
B i r g e r m o a s t e r

Sequence-based String Similarity

- Length of longest common prefix (PREFIX)
- Longest common subsequence ratio (LCSR)[1]

$$\mathrm{LCSR}(x, y) = \frac{|\,\mathrm{Brgermster}\,|}{\max(|x|, |y|)} = 10 / 13$$

B ü r g e r m e i s t e r
B i r g e r m o a s t e r

[1] *Bitext maps and alignment via pattern recognition (Milamed, 1999)*

Sequence-based String Similarity

- Length of longest common prefix (PREFIX)
- Longest common subsequence ratio (LCSR)
  - LCSR between bigram (BI-SIM) sequences.[1]

Bi-Sim (x, y) = 7 / 12

Bü	ür	rg	ge	er	rm	me	ei	si	st	te	er
Bi	ir	rg	ge	er	rm	mo	oa	as	st	te	er

[1] *Identification of confusable drug names: A new approach and evaluation methodology (Kondrak and Dorr, 2004)*

Sequence-based String Similarity

- Length of longest common prefix (PREFIX)
- Longest common subsequence ratio (LCSR)
  - LCSR between bigram (BI-SIM) and trigram (TRI-SIM) sequences.[1]

Tri-Sim (x, y) = 5 / 11

Bür	ürg	rge	ger	erm	rme	mei	eis	sit	ste	ter
Bir	irg	rge	ger	erm	rmo	moa	oas	ast	ste	ter

[1] *Identification of confusable drug names: A new approach and evaluation methodology (Kondrak and Dorr, 2004)*

Sequence-based String Similarity

- Length of longest common prefix (PREFIX)
- Longest common subsequence ratio (LCSR)
  - LCSR between bigram (BI-SIM) and trigram (TRI-SIM) sequences.
- Length-normalized edit distance (NED)[1,2]

[1] *Binary codes capable of correcting deletions, insertions and reversals (Levenshtein, 1966)*
[2] *Identification of confusable drug names: A new approach and evaluation methodology (Kondrak and Dorr, 2004)*

Sequence-based String Similarity

- Length of longest common prefix (PREFIX)
- Longest common subsequence ratio (LCSR)
  - LCSR between bigram (BI-SIM) and trigram (TRI-SIM) sequences.
- Length-normalized edit distance (NED)[1,2]
  - NED between bigram (BI-DIST) and trigram (TRI-DIST) sequences.

[1] *Binary codes capable of correcting deletions, insertions and reversals (Levenshtein, 1966)*
[2] *Identification of confusable drug names: A new approach and evaluation methodology (Kondrak and Dorr, 2004)*

Sequence-based String Similarity

- Length of longest common prefix (PREFIX)
- Longest common subsequence ratio (LCSR)
  - LCSR between bigram (BI-SIM) and trigram (TRI-SIM) sequences.
- Length-normalized edit distance (NED)
  - NED between bigram (BI-DIST) and trigram (TRI-DIST) sequences.
- Edit distance between phonetic codes (cologne phonetics)[1]

Bürgermeister	—>	17476827
Birgermoaster	—>	17476827

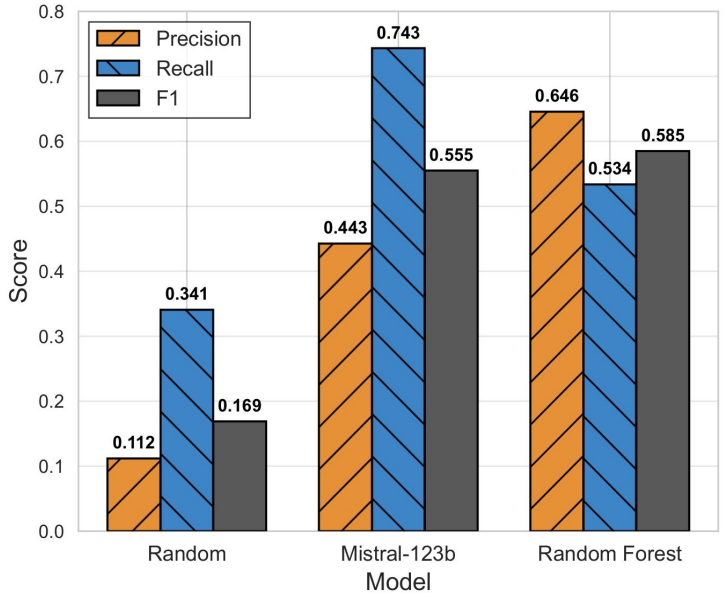
[1] *Die Kölner Phonetik - Ein Verfahren zur Identifizierung von Personennamen auf der Grundlage der Gestaltanalyse (Postel, 1969)*

Overview

- Introduction
- Method
- Results
- Conclusion

Word-Pair Classification

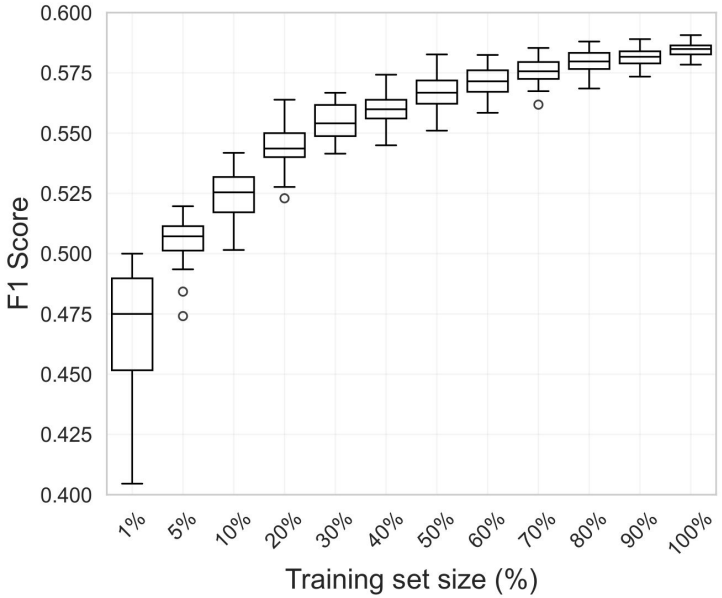
- Reference: best-performing LLM from prior work.[1]
- Mistral-123b[1] excels in recall (0.743) but suffers in precision (0.443).
- Random forests achieve the best overall classification performance (F1=0.585).



[1]*Make Every Letter Count: Building Dialect Variation Dictionaries from Monolingual Corpora (Litschko et al., 2025)*

Effect of Training Set Size

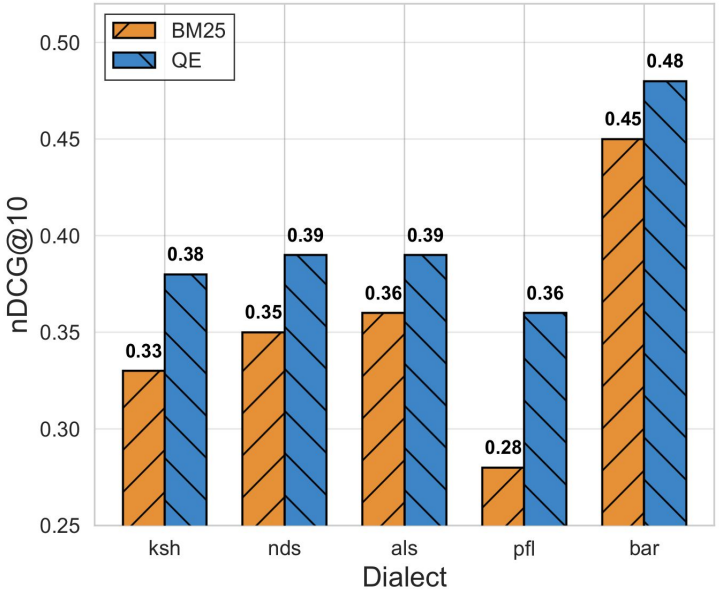
- Training on 10% of the training data (8k instances) yields competitive results (avg. F1 = 0.52).
- Training on 40% of the training data (avg. F1=0.56) outperforms Mistral.
- Adding more training data yields only marginal improvments.



(Result shown for 40 different random seeds.)

Cross-Dialect Information Retrieval

- Query expansion (QE) improves retrieval results:
  - +0.05 nDCG@10 (+14.7%)
  - +0.08 Recall@100 (+25.1%)
- Overall, ~51% of all queries contain keywords covered in our dictionary (see paper).



Overview

- Introduction
- Method
- Results
- Conclusion

Conclusion

- We release dialect variation dictionaries for 5 German dialects.
- Word-pair classification: Random forest are more efficient (lower training/inference cost) and more effective than LLMs.
- Dialect retrieval: Query expansion using our dictionaries consistently improves BM25 performance.
- We make our code and data available for future uptake.

Thank you!

GitHub