

Does a Lexicon-Trained Stemmer Generalize?

A 20-Language Study of Held-Out Lexical Families

Leo Galambos

Abstract—A stemmer induced from lexical evidence is useful beyond exact dictionary membership only if training-derived transformations transfer to surfaces that were not used to construct the model. This study measures that property for Radixor, a current implementation of the Egothor patch-command stemming lineage, across 20 UniMorph-derived language resources. Dictionary rows are held out atomically as lexical families. For each resource, five protocol-fixed deterministic family rankings define nested 10%–100% training prefixes. The primary test scope removes every held-out occurrence whose lowercased, parser-normalized surface form appeared in training. The protocol contains 1,000 build/evaluation scenarios, of which 900 have a non-empty held-out population. The headline metric is exact recovery on unseen forms whose resource-defined canonical target differs from the input. The resource-macro value is 60.49% at 10% training-family coverage and 69.07% at 90%; the corresponding across-resource medians are 63.72% and 70.44%. These values provide direct operational evidence that training-derived patch commands are successfully applied beyond the exact training-surface set. Because each coverage level evaluates the complement of its own training prefix, however, the test population changes with coverage: endpoint differences are descriptive coverage-conditioned contrasts, not causal estimates of the effect of adding training families. The claim is further bounded to within-resource transfer from predominantly inflectional lexical evidence and is not evidence of out-of-domain accuracy, contextual lemmatization ability, universal stemming quality, or causal differences among languages.

Index Terms—stemming, morphology, information retrieval, text normalization, lexical resources, generalization, multilingual NLP, UniMorph

I. INTRODUCTION

STEMMING maps related surface forms to a common indexing representation and remains an established normalization option in information retrieval (IR) and search-oriented text processing [1], [2]. Classical stemmers encode language knowledge directly as affix rules or related heuristics. Lovins’ context-sensitive longest-match algorithm [3] and Porter’s suffix-stripping algorithm [4] established influential rule-driven approaches. Other systems combine morphological reasoning with lexical evidence, as in Krovetz’s conservative inference-oriented method [5]. In IR, the objective is typically useful conflation rather than recovery of a uniquely privileged linguistic lemma [6]–[8].

A different design is to induce transformation behavior from examples. Corpus-driven methods such as YASS [9] and HPS [10] infer regularities from observed data. Trainable lemmatizers have likewise learned transformations from full-form–lemma pairs and evaluated OOV behavior [11], [12]. Modern contextual systems such as Lematus explicitly use

sentence context to address unseen and ambiguous forms [13], while later work augments contextual lemmatization with UniMorph inflection tables [14]. Radixor addresses a narrower, context-free normalization setting relevant to deterministic search pipelines. The Egothor method occupies a supervised lexical position in this design space: a dictionary supplies canonical targets and associated variants, edit-derived patch commands are learned from those mappings, and the commands are organized in a trie [15]. The lexical resource is therefore evidence used to construct a transformation model, not merely a table queried at runtime.

That distinction is empirical, not terminological. If a lexicon-trained system succeeds only for surfaces observed during construction, then its open-vocabulary value is limited. If a model built from one set of dictionary families can correctly transform forms from families excluded from training, it has operationally reused training-derived behavior outside the exact training-surface set. This matters in search systems because maintained lexical resources are finite whereas indexed and queried vocabularies are open-ended.

The underlying question is not new. The original Egothor work already measured errors after hiding part of a dictionary [15]; later work explicitly examined generalization in semi-automatic stemmers [16]. Stempel, a direct descendant of the Egothor implementation, used shuffled inflection sets, nested training samples, and a separate test portion containing only words unseen during training [17]. HPS also evaluated previously unseen words across six languages [10]. The contribution of the present study is therefore not the invention of an unseen-word or family-holdout experiment. It is a controlled multilingual extension of this line of evaluation: 20 checked-in lexical resources, five protocol-fixed family rankings per resource, a common 10%–100% coverage grid, explicit normalized-surface leakage control, and artifact-level provenance for every raw observation.

We ask three research questions:

- **RQ1:** Does a model induced from a subset of dictionary families recover canonical targets for changed surface forms belonging to families excluded from training?
- **RQ2:** How does held-out exactness vary across training-family coverage conditions from 10% to 90%, when each condition is evaluated on its own unselected complement?
- **RQ3:** How heterogeneous are these coverage-conditioned curves across the 20 language-resource combinations?

The study contributes (i) a reproducible multilingual family-holdout protocol with explicit exact-surface overlap removal; (ii) 1,000 deterministic build/evaluation scenarios, including 900 scenarios with non-empty held-out generalization populations; and (iii) descriptive resource-level and macro evidence

that separates the central transfer claim from stronger claims the experiment cannot support. The coverage sweep estimates performance on the remaining unselected complement at each training level; unlike a fixed-test learning-curve design, it does not isolate the marginal effect of additional training data. Exact canonical-target recovery is intentionally used here as an operational measure of transformation transfer, not as a universal measure of IR effectiveness or stemming quality.

II. BACKGROUND AND RELATED WORK

A. Stemming and Evaluation

Stemming algorithms vary in both linguistic ambition and evaluation objective. Lovins [3] and Porter [4] demonstrated that manually encoded affix transformations can generalize naturally to unseen tokens because their rules are not tied to lexical membership. Krovetz [5] illustrated a more conservative design in which lexical evidence informs morphological inference.

Evaluation is correspondingly multidimensional. Paice [6] formalized intrinsic under-stemming and over-stemming measures over predefined groups. Hull [7] argued that aggregate retrieval scores can conceal substantial behavior and emphasized detailed statistical analysis. Weissweiler and Fraser [8] compared German stemmers against two purpose-built gold standards and reiterated that an IR stemmer need not emit a linguistically exact lemma to produce useful conflation. These results motivate keeping the present exact-target transfer question separate from pairwise conflation quality, retrieval effectiveness, and runtime.

B. Learned Stemming and OOV Transformation

Data-driven stemming and lemmatization have repeatedly addressed the problem of unseen forms. YASS clusters a collection lexicon using string-distance measures [9]. HPS uses an unsupervised first stage to create training data for a maximum-entropy classifier and explicitly reports results on previously unseen words in six languages [10]. Kanis and Müller [11] constructed a lemmatizer from full-form-lemma dictionaries with dedicated mechanisms for OOV forms. Jongejan and Dalianis [12] automatically learned lemmatization rules handling changes in prefixes, infixes, and suffixes across ten languages. Contextual neural lemmatization extends the problem further: Lematus evaluates unseen and ambiguous forms in 20 languages and finds sentence context particularly useful for ambiguity [13]; subsequent work combines such models with UniMorph inflection tables [14]. The present study is intentionally context-free and therefore addresses transformation transfer rather than contextual disambiguation.

The Egothor lineage is supervised and transformation-centric. Galambos [15] derives compact patch commands from word-to-target pairs using minimum-cost edit paths and stores those commands in a trie. The 2001 evaluation hid part of a dictionary before trie construction, and follow-up work [16] compared semi-automatic stemmers that learn from transformation dictionaries. Stempel later applied the same algorithmic lineage to Polish and documented a particularly relevant protocol: approximately 69,000 inflection sets were

shuffled, nested training samples were drawn from one portion, and a separate fixed portion was reserved for testing only forms unseen during training [17]. That fixed-test construction isolates training-sample size more directly than the coverage-conditioned complement used in the present study.

The present protocol should therefore be read as a breadth and control extension of prior set-holdout work rather than a claim of methodological priority or a stronger causal learning-curve design. Its distinguishing features are the 20-language scope, five protocol-fixed deterministic rankings per resource, a common percentage-based coverage schedule, exact parser-normalized surface-overlap exclusion, and checked-in raw counters with source/model checksums.

C. Lexical Resource Context

The 20 default inputs evaluated here are attributed by the Radixor model catalog to UniMorph-derived sources. UniMorph provides normalized inflection tables and a language-independent morphological feature schema across many languages [18]; it has also been used as type-level lexical supervision for neural lemmatization [14]. Consequently, the evidence in this study is predominantly inflectional rather than a balanced sample of inflectional and derivational morphology. The experimental inputs are not claimed to be byte-identical to UniMorph 4.0: the Radixor provenance explicitly records that immutable upstream revisions were not retained for these legacy imports [19]. The checked-in compressed model inputs and their SHA-256 digests are therefore the authoritative experimental data.

III. TRANSFORMATION MODEL UNDER STUDY

Radixor learns a patch command for each observed surface-to-canonical mapping. In the historical formulation [15], a command is derived from a minimum-cost edit path and can encode deletion, insertion, replacement, and preserved matching text. A simple mapping such as

$$\text{running} \rightarrow \text{run} \quad (1)$$

can be encoded by a command that removes the four final characters. The relevant property for this study is semantic: the stored value describes *how* to transform an input rather than storing only a final output string for a single key.

Training-derived commands are organized in a backward-traversed trie and compiled into the immutable representation used by the measured Java implementation. Structural reduction can make a preferred command available before every input character has been consumed, allowing the same training-derived command to be selected for an input surface that was never inserted as a training key [19]. The experiment uses this complete construction path rather than an isolated encoder micro-model.

This yields an operationally precise interpretation of the headline metric. The experimental builder inserts commands derived only from selected training rows. If a normalized surface w is absent from those rows, $w \neq c$, and the model nevertheless returns canonical target c , the success necessarily results from applying behavior contained in the training-induced

trie to a surface outside the exact training set. The experiment does not identify which linguistic generalization caused each success, but a changed-form exact success cannot be explained by exact training-surface membership.

IV. LEXICAL RESOURCES AND TERMINOLOGY

A parsed Radixor dictionary row contains one canonical target followed by associated variants. We call such a row a *dictionary family* or *lexical family*. This terminology is intentionally weaker than asserting that every row is a complete, disjoint linguistic paradigm: lexical sources may contain homography, duplicate analyses, extraction artifacts, or overlapping assignments. In UniMorph-derived inputs the canonical target is typically lemma-like or a designated base form, but this article refers to it as the *resource-defined canonical target* because the experiment evaluates the stored mapping rather than asserting an independent linguistic gold lemma. The surface-overlap control in Section V-B is necessary precisely because a normalized surface can occur in more than one row.

The production parser trims columns and lowercases dictionary content with `Locale.ROOT`; fields containing Unicode whitespace are not accepted. All membership checks and exact comparisons in this experiment therefore operate on the same parser-normalized strings used to build the trie [19].

Table I summarizes the 20 default inputs. Together they contain 767,167 dictionary families and 7,454,592 parsed form occurrences; 5,967,906 occurrences differ from their row’s canonical target and therefore belong to the changed-form population. Resource size varies by several orders of magnitude, from 69 Persian families to 396,939 English families. The primary resource-macro summary weights each configured language resource equally; it therefore answers a different descriptive question from a token-pooled micro average, which would be dominated by the largest resources. Per-resource results and robust across-resource summaries are retained so that no single aggregation is treated as canonical.

V. EXPERIMENTAL METHODOLOGY

A. Frozen Family-Level Splits

The experiment uses protocol `radixor-generalization-v1`. Let

$$F_i = (c_i, V_i) \quad (2)$$

denote one parsed dictionary family with canonical target c_i and normalized variants V_i . A row is atomic: it is either fully selected for training or fully withheld.

For every model and seed, rows are ordered by a stable 64-bit FNV-derived hash over the protocol identifier, model ID, seed, canonical target, and variants. Physical line number is used only as a deterministic tie-breaker. Requested coverage levels define nested 10%, 20%, ..., 100% prefixes; within one seed, every row selected at a lower coverage remains selected at all higher levels. The frozen protocol uses five seeds recorded in the experiment source:

$$\begin{aligned} &2654435761, \quad 2611923443488327891, \\ &7046029254386353131, \quad 11400714819323198485, \\ &15111065706836454659. \end{aligned} \quad (3)$$

At each level the experiment constructs the same contracted compiled patch-command trie used by the released Java model path. As an implementation-validity guard, the application first builds a full-coverage experimental trie and compares its output with the default model loader on every parsed canonical target and variant; any discrepancy aborts the run [19]. The complete design contains $20 \times 10 \times 5 = 1,000$ build/evaluation scenarios. Because 100% coverage has no withheld population, $20 \times 9 \times 5 = 900$ scenarios contribute non-empty held-out generalization measurements. Figure 1 summarizes the protocol.

B. Evaluation Scopes and Surface Leakage Control

Three scopes are retained in the raw counters. *Whole dictionary* combines trained and withheld rows and is therefore not isolated generalization evidence. *Withheld rows* contains every occurrence from rows outside the training prefix. The primary *unseen surface* scope begins with withheld rows and excludes an occurrence if the identical parser-normalized surface occurs in any selected training row.

Let T be the selected family set and

$$S_T = \{w : w \in V_i \cup \{c_i\}, F_i \in T\} \quad (4)$$

be the set of parser-normalized training surfaces. A withheld occurrence (w, c_i) enters the primary test population U_T only if

$$F_i \notin T \quad \wedge \quad w \notin S_T. \quad (5)$$

The filter removes exact surface leakage, not morphological relatedness. Held-out families may share productive transformations with training families; measuring whether the compiled model can exploit such reusable structure is the point of the experiment.

C. Metrics

Let $g_T(w)$ be the preferred output of the trie built from training families T . For the primary unseen population U_T , the report stores three exact-target rates:

$$A_{\text{all}} = \frac{\sum_{(w,c) \in U_T} \mathbf{1}[g_T(w) = c]}{|U_T|}, \quad (6)$$

$$A_{\text{chg}} = \frac{\sum_{(w,c) \in U_T, w \neq c} \mathbf{1}[g_T(w) = c]}{|\{(w,c) \in U_T : w \neq c\}|}, \quad (7)$$

$$A_{\text{pres}} = \frac{\sum_{(w,c) \in U_T, w=c} \mathbf{1}[g_T(w) = c]}{|\{(w,c) \in U_T : w = c\}|}. \quad (8)$$

We refer to these as *all exact*, *changed exact*, and *canonical preserved*. Changed exact is the headline transfer measure because an identity output cannot score as correct when $w \neq c$. For a membership-only lookup with identity fallback, changed exact on U_T would therefore be 0% by construction. Radixor is not evaluated against that reference as a competitive stemmer; the reference simply clarifies what the headline metric establishes operationally.

At 100% coverage there is no withheld population, so unseen metrics are undefined rather than zero. Generalization curves are consequently discussed over 10%–90% coverage.

TABLE I

EXPERIMENTAL LEXICAL RESOURCES IN THE 20 DEFAULT RADIXOR MODELS. “CHANGED” COUNTS PARSED OCCURRENCES WHOSE SURFACE DIFFERS FROM THE ROW’S CANONICAL TARGET. THE MODEL CATALOG ATTRIBUTES ALL DEFAULT INPUTS TO UNIMORPH-DERIVED SOURCES; EXACT LEGACY UPSTREAM REVISIONS WERE NOT RETAINED.

Language	Model	Families	Forms	Changed	Language	Model	Families	Forms	Changed
Czech	cs-cz-default	5,113	56,612	46,563	Norwegian Bokmål	nb-no-default	17,929	90,757	57,381
Danish	da-dk-default	4,179	32,256	23,900	Norwegian Nynorsk	nn-no-default	4,688	19,651	13,562
Dutch	nl-nl-default	4,992	31,466	21,485	Persian	fa-ir-default	69	3,770	3,632
English	us-uk-default	396,939	1,002,414	208,540	Polish	pl-pl-unimorph	9,990	132,308	112,351
Finnish	fi-fi-default	57,027	1,865,215	1,754,690	Portuguese	pt-pt-default	4,001	215,490	207,488
French	fr-fr-default	59,240	474,110	365,969	Russian	ru-ru-default	37,410	806,279	731,471
German	de-de-default	54,092	333,036	242,501	Spanish	es-es-default	65,059	926,393	806,272
Hebrew	he-il-default	2,358	61,071	56,356	Swedish	sv-se-default	12,371	110,468	85,737
Hungarian	hu-hu-default	19,406	935,713	896,938	Ukrainian	uk-ua-default	1,493	15,737	12,752
Italian	it-it-default	10,009	337,546	317,542	Yiddish	yi-default	802	4,300	2,776
Total		767,167	7,454,592	5,967,906					

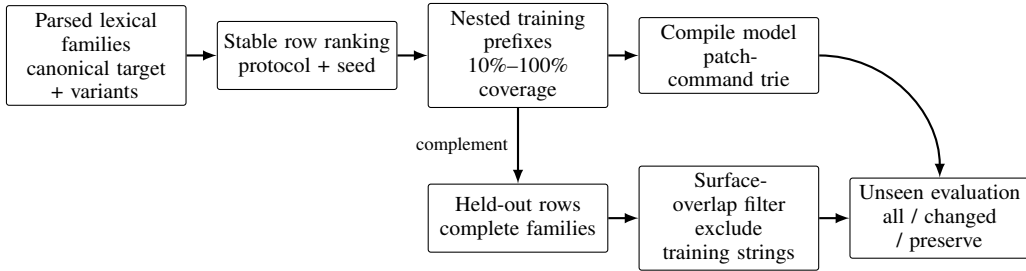


Fig. 1. Coverage-conditioned family-holdout protocol. A parsed dictionary row is atomic: its canonical target and associated variants are selected or withheld together. The primary unseen scope additionally removes a held-out occurrence if the identical parser-normalized surface occurs in training.

D. Coverage Conditions and Changing Evaluation Populations

The nested prefixes make training sets comparable by inclusion within each seed, but their held-out complements are not fixed. At 10% coverage the unseen scope is drawn from roughly the remaining 90% of families after surface-overlap exclusion; at 90% coverage it is drawn from roughly the remaining 10%. Thus the numerator and denominator at different coverage levels refer to different sets of items. The surface-overlap filter can change the composition further because its exclusions depend on the current training prefix.

Accordingly, the x-axis in this study defines *coverage-conditioned generalization*: performance of the model trained at coverage p on the unseen portion of that condition’s unselected complement. Differences between coverage points combine a changed training set with a changed evaluation population and must not be interpreted as an isolated treatment effect of adding training families. A fixed anchor test set, as in the historical Stempel experiment [17], would be required to estimate that effect more directly.

E. Aggregation and Split Sensitivity

For each resource and coverage level, the primary summary statistic is the median across the five protocol-fixed splits. The repository also preserves the observed split minimum and maximum. These ranges are sensitivity summaries for the five protocol-fixed deterministic family rankings; they are not confidence intervals and no population-level random-sampling model is assumed.

The resource-macro curve first computes the five-split median separately for each resource and then takes the arithmetic mean

over the 20 resource medians. This gives each configured language resource equal weight; it is an explicit descriptive estimand rather than an estimate for a randomly sampled language population. Because a mean of heterogeneous resource medians can itself hide distributional structure, the revised analysis additionally reports the across-resource median and interquartile range (IQR) at the 10% and 90% endpoints. The 25th and 75th percentiles use linear interpolation between adjacent ordered resource values. All cross-language statistics in this paper are descriptive; the 20 resources are not treated as a random sample from a population of languages.

F. Why Competitor Stemmers Are Not a Coverage Baseline

The manipulated variable in this experiment is the amount and identity of *Radixor training evidence*. Fixed rule stemmers, closed lexical analyzers, and independently trained statistical stemmers do not expose a commensurate control in which the same selected family prefixes can be supplied while holding their remaining algorithm constant. Adding their ordinary outputs to the same plot would therefore confound a within-system coverage-sweep question with overall stemmer quality. Competitor comparisons are intentionally outside this study. The membership-only 0% changed-exact reference above is used only to define the minimum capability implied by successful transfer, not to rank systems.

VI. RESULTS

A. RQ1: Transfer Beyond the Training Surface Set

RQ1 is positive for the evaluated resources at the aggregate level. At 10% training-family coverage, the resource-macro

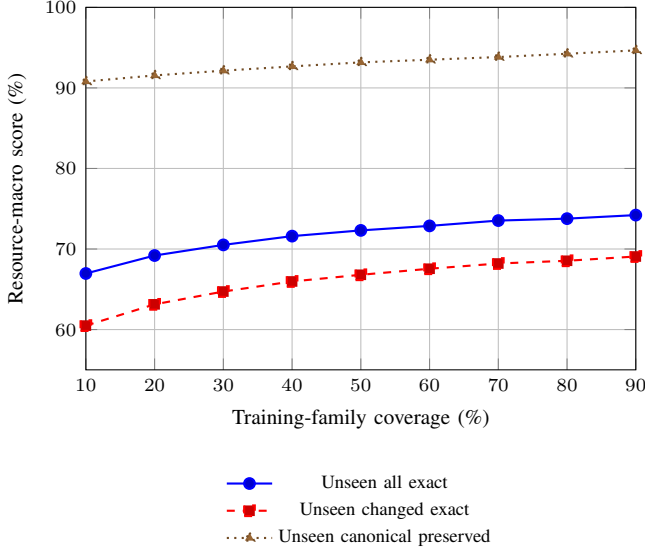


Fig. 2. Resource-macro coverage-conditioned generalization across 20 resources. Each point is the arithmetic mean of the 20 per-resource five-split medians. The primary measure, unseen changed exact, excludes occurrences for which the normalized input already equals its canonical target. Each coverage level evaluates its own unselected complement, so the x-axis is not a fixed-test estimate of the causal effect of additional training families.

unseen changed-exact rate is 60.49%; the across-resource median is 63.72% (IQR 50.65–76.67%). These are changed forms whose exact normalized surface was excluded from training. A correct output therefore requires the training-induced trie to select a non-identity transformation that maps an unseen surface to the resource-defined canonical target.

The result is stronger than a statement that the runtime accepts unknown strings: it measures exact resource-target recovery on them. It is weaker than a claim of linguistically correct OOV stemming in arbitrary text. The canonical target, family organization, and test distribution all come from the same checked-in lexical resource.

At 90% coverage, the resource-macro changed-exact rate is 69.07% and the across-resource median is 70.44% (IQR 64.20–82.10%). Thus the central observation—successful application of training-derived behavior to surfaces outside the exact training set—persists across the coverage range.

B. RQ2: Coverage-Conditioned Curves and Endpoint Contrasts

Figure 2 shows the resource-macro coverage-conditioned curves. Changed exact is 60.49% at 10% training coverage, 63.13% at 20%, 66.80% at 50%, and 69.07% at 90%. The 90%–10% resource-macro endpoint difference is 8.574 pp. Unseen all exact changes from 66.96% to 74.21% across the same endpoint conditions, while canonical preservation changes from 90.80% to 94.66%. Because each point uses its own held-out complement, these differences summarize coverage conditions; they do not isolate the causal effect of additional training families.

Table II reports a robust companion summary for the headline metric. The across-resource median 90%–10% endpoint difference is 7.764 pp, with an IQR of 5.679–10.614 pp. Every

TABLE II
ACROSS-RESOURCE SUMMARY FOR UNSEEN CHANGED EXACT. ENDPOINT VALUES ARE PERCENTAGES; Δ IS THE PER-RESOURCE 90% MINUS 10% DESCRIPTIVE ENDPOINT DIFFERENCE IN PERCENTAGE POINTS. THE TWO ENDPOINT CONDITIONS EVALUATE DIFFERENT HELD-OUT COMPLEMENTS.

Statistic	10%	90%	Δ
Resource-macro mean	60.491	69.065	8.574
Across-resource median	63.718	70.442	7.764
Across-resource IQR	50.649–76.669	64.195–82.102	5.679–10.614

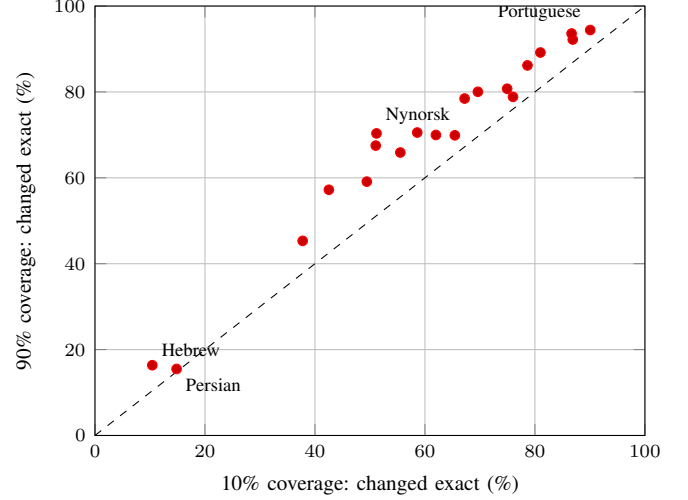


Fig. 3. Per-resource median unseen changed exact at 10% versus 90% training-family coverage. The dashed identity line marks equal endpoint values. All 20 resources lie above it; the descriptive endpoint differences range from 0.651 to 19.147 percentage points. The endpoints evaluate different held-out complements and therefore do not estimate a common training-data effect.

one of the 20 resource medians is higher in the 90% condition than in the 10% condition. This 20/20 pattern is descriptive only: the resources were not sampled randomly, and the two endpoints evaluate different complements, so the pattern is neither a population-level significance result nor an estimate of a common training-data effect.

The resource-macro curve is numerically flatter at higher coverage, but this experiment alone does not establish *why*. The pattern may reflect transformation reuse, changed command ranking, the changing composition of the held-out complement, or a combination of these effects. Distinguishing those mechanisms requires a fixed-test or command-inventory analysis and is therefore outside the claims of this experiment.

C. RQ3: Cross-Resource Heterogeneity

The resource-macro mean is not a universal profile. Table III lists the median changed-exact endpoints for all 20 resources. At 10%, values range from 10.423% for Hebrew to 90.053% for Portuguese. At 90%, they range from 15.493% for Persian to 94.412% for Portuguese. Endpoint differences range from 0.651 pp for Persian to 19.147 pp for Norwegian Nynorsk.

Figure 3 shows all 20 endpoint pairs against the identity line. The fact that every point lies above the line is a compact representation of the positive 10%–90% endpoint change without hiding the large level differences among resources.

TABLE III

PER-RESOURCE MEDIAN UNSEEN CHANGED EXACT AT 10% AND 90% TRAINING-FAMILY COVERAGE. Δ IS THE DESCRIPTIVE 90% MINUS 10% ENDPOINT DIFFERENCE IN PERCENTAGE POINTS; THE ENDPOINTS EVALUATE DIFFERENT HELD-OUT COMPLEMENTS. VALUES DESCRIBE SPECIFIC CHECKED-IN LANGUAGE-RESOURCE-MODEL COMBINATIONS AND MUST NOT BE INTERPRETED AS CAUSAL RANKINGS OF LANGUAGES OR MORPHOLOGICAL COMPLEXITY.

Language	10%	90%	Δ	Language	10%	90%	Δ
Czech	65.439	69.900	+4.461	Norwegian Bokmål	58.598	70.546	+11.948
Danish	51.055	67.504	+16.449	Norwegian Nynorsk	51.191	70.338	+19.147
Dutch	55.524	65.891	+10.367	Persian	14.842	15.493	+0.651
English	76.010	78.854	+2.844	Polish	61.996	69.972	+7.976
Finnish	78.644	86.177	+7.533	Portuguese	90.053	94.412	+4.359
French	69.627	80.031	+10.404	Russian	74.948	80.744	+5.796
German	49.432	59.108	+9.676	Spanish	80.995	89.176	+8.181
Hebrew	10.423	16.362	+5.939	Swedish	67.214	78.458	+11.244
Hungarian	86.876	92.203	+5.327	Ukrainian	42.530	57.218	+14.688
Italian	86.665	93.600	+6.935	Yiddish	37.767	45.318	+7.551

Intermediate coverage-conditioned behavior also differs. Figure 4 shows four contrasting curves and the observed minimum–maximum range across the five protocol-fixed splits at each level. Portuguese begins high and changes modestly. Norwegian Nynorsk shows a large positive endpoint contrast. English, despite being the largest resource by family count, has a comparatively small endpoint difference. Hebrew remains low throughout. The purpose of these examples is to expose different coverage-conditioned shapes and split sensitivity; the complete 20-resource endpoints remain in Table III.

The variation cannot be attributed directly to language typology. Resource size, paradigm composition, ambiguity, canonical-target conventions, orthography, extraction quality, and compatibility with the transformation/trie representation are all potential confounders. Persian illustrates the danger of over-interpretation: its resource has only 69 families, so 10% training contains seven rows and 90% contains 62. Conversely, high transfer in Portuguese, Italian, or Hungarian does not isolate linguistic regularity from resource construction. The defensible RQ3 conclusion is therefore that transfer is highly heterogeneous across the evaluated language-resource-model combinations.

VII. DISCUSSION

A. What the Experiment Establishes

The experiment establishes a narrow but concrete capability. The compiled model frequently returns the resource-defined canonical target for a changed surface that belongs to a withheld dictionary row and whose exact parser-normalized string was absent from training. Since the trie contains only commands derived from selected training rows, such a success is an application of training-induced behavior beyond exact training-surface membership.

This is the strongest interpretation supported by the design. It does not require a claim that the model has inferred an explicit linguistic rule, discovered a morpheme boundary, or learned the morphology of a language in a general sense. A correct unseen transformation can arise because input structure causes an already learned patch command to be selected in a new context. The experiment measures that operational transfer directly.

B. What the Coverage Curves Do Not Establish

The resource-macro endpoint difference is consistent with substantial reuse of transformations across families, but the coverage sweep changes both the training prefix and the evaluated complement. It therefore cannot by itself attribute the numerical difference to added training evidence, establish command-set saturation, or show that later dictionary rows merely reinforce known commands. Those mechanisms require a fixed-test experiment or separate command-inventory analysis.

The same discipline applies to resource size. English is large and has a shallow endpoint difference, while some smaller resources show larger contrasts, but these observations do not establish a causal relationship between dictionary size and coverage-curve shape. Any such analysis would require a prespecified cross-resource model together with controls for resource construction, ambiguity, and linguistic properties.

C. Implications for Search and NLP Systems

For search engineering, the result distinguishes a compiled transformation model from pure exact lexical membership. Finite lexical evidence can support deterministic transformations for additional surfaces without requiring the queried surface itself to have been inserted during training. This property is relevant when query and index vocabularies exceed a maintained lexical resource.

For NLP evaluation, the experiment also illustrates why “unseen” must be defined carefully in lexical datasets. Withholding a row is insufficient when the same normalized surface can occur in another training row. Exact surface-overlap filtering is a straightforward guard against that leakage mode, although it does not and should not remove shared morphology between families. Conversely, a context-free stemmer cannot resolve genuine target ambiguity from sentence context; this study should therefore not be read as a substitute for contextual lemmatization [13].

D. Exact Target Recovery Versus Stemming Quality

Exact canonical-target recovery is appropriate for the present question because it tests whether a learned transformation reproduces a known resource mapping on an unseen surface. It is not a universal definition of good stemming. Two different output strings can induce the same useful retrieval conflation,

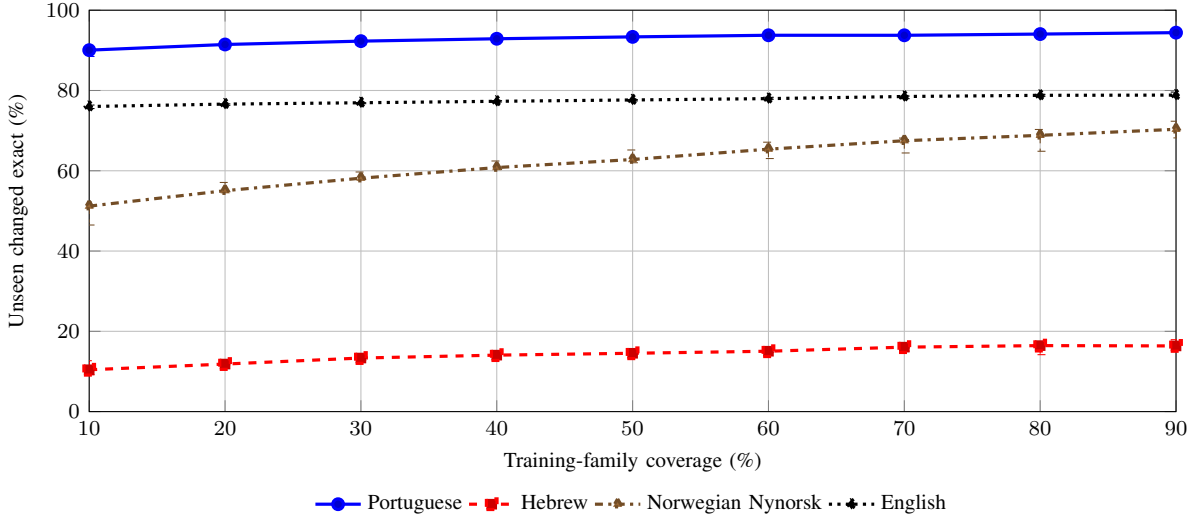


Fig. 4. Contrasting coverage-conditioned curves with observed minimum–maximum ranges across the five protocol-fixed splits. Portuguese is high from low coverage; Hebrew remains low; Norwegian Nynorsk has the largest 10%–90% median endpoint difference; and English is the largest resource by family count. Each coverage level has a different held-out complement. Error bars are descriptive split ranges, not confidence intervals.

and a linguistically valid lemma can still be suboptimal for a particular retrieval task [6]–[8]. The study therefore makes no comparative quality claim against rule stemmers, morphological analyzers, or other learned systems. Pairwise conflation quality, over-/under-stemming, downstream retrieval effectiveness, index size, and runtime are distinct evaluation dimensions.

VIII. THREATS TO VALIDITY

A. Within-Resource Rather Than Out-of-Domain Generalization

Training and evaluation rows come from the same checked-in lexical resource for each model. The design controls row membership and exact parser-normalized surface overlap, but it does not test another domain, different annotation conventions, misspellings, arbitrary compounds, unseen scripts, social-media noise, or novel named entities. Related withheld families can share productive morphology with training families; that sharing is the intended object of measurement. Accordingly, *generalization* in this paper always means within-resource, out-of-training-family transfer.

B. Dictionary Families Are Not Guaranteed Linguistic Partitions

A dictionary row is the atomic unit of the experiment, but rows are not assumed to form a perfect partition of linguistic lexemes or paradigms. The same surface can be associated with multiple rows, and source transformations may reflect extraction and annotation conventions. The exact-surface filter removes the most direct training/test overlap but does not make held-out rows semantically independent. If one parser-normalized surface occurs only in held-out rows under multiple distinct canonical targets, all such occurrences remain in the unseen scope; a deterministic context-free primary output cannot satisfy conflicting targets simultaneously. Such ambiguity therefore

imposes an exact-match ceiling that is not equivalent to a failure of transformation reuse. For this reason the title and analysis use *lexical families* rather than claiming complete morphological paradigms.

C. Lexical-Resource Provenance and Annotation

The model catalog attributes all 20 default inputs to UniMorph-derived sources, but exact immutable upstream revisions were not retained for the legacy imports. The actual experiment remains byte-identifiable because each raw row records the independently versioned model and SHA-256 of the compressed input. Reconstructing every input from a historical upstream commit, however, is not possible from the retained metadata.

The experiment inherits each resource’s canonical-target choices, duplicate analyses, and extraction quality. Because the default inputs are UniMorph-derived inflection tables, the study predominantly tests transfer across inflectional evidence and should not be generalized to derivational stemming without separate data. Cross-resource differences are therefore properties of specific language-resource-model combinations, not estimates of intrinsic morphological difficulty.

D. Coverage and Evaluation-Population Coupling

The coverage sweep is not a fixed-test learning curve. As the training prefix grows, the held-out complement shrinks and changes composition; surface-overlap exclusions are also coverage-dependent. Consequently, a 90%–10% endpoint contrast conflates differences in the trained model with differences in the evaluated items. The article therefore treats coverage points as separate descriptive conditions and does not estimate a causal marginal return to additional families. A future fixed-anchor test protocol could answer that different question directly.

E. Resource Imbalance and Small Samples

Family counts range from 69 to 396,939. Resource-macro aggregation prevents the largest resources from dominating the central summary, but it cannot make small resources equally informative. Persian and Yiddish have coarse coverage increments and visibly wider split ranges than many large resources. The added median/IQR summary reduces dependence on the resource-macro mean but does not eliminate the underlying imbalance.

F. Five Deterministic Rankings and Descriptive Uncertainty

Five seeded hash rankings provide a reproducible sensitivity probe, not a statistical sampling distribution. The observed minimum–maximum ranges shown for selected resources are not confidence intervals. The study consequently avoids p-values and population-level claims over languages. A larger resampling study could estimate sensitivity to many more family orderings, particularly for small resources, but it is not required for the deterministic descriptive claim reported here.

G. Model-Class and Context Limitations

The measured construction uses the same backward-traversed patch-command trie class across all evaluated languages. This gives the system a particular inductive bias toward transformations recoverable from reversed surface structure. Prefixal, non-concatenative, or highly ambiguous phenomena may interact differently with that representation; low or high scores for any language cannot identify such mechanisms causally. The model is also context-free, unlike modern contextual lemmatizers that can use surrounding tokens to disambiguate a surface form [13].

H. Metric Scope

Changed exact removes no-op inflation but remains a lexical exact-match measure. It does not directly measure semantic conflation, index compression, recall, ranking effectiveness, or user utility. Canonical preservation captures one failure mode—changing a form that should remain unchanged under the resource mapping—but is not equivalent to general overstemming control.

I. Implementation and Measurement State

The article-facing repository snapshot and the physical measurement source are intentionally distinguished. The public state cited by this article is commit 0c3b13f, which contains the frozen results, raw counters, checksums, and documentation. The generalization run identifies Radixor/Java version 4.2.0-6-g84e57fb, core revision 84e57fb, and source state dirty. The repository records a SHA-256 manifest of the measured source state and a SHA-256 of the generator. Thus “dirty” denotes a captured, auditable source state rather than an unspecified one. This provenance is more precise than retroactively assigning the measurements to the later documentation commit.

IX. REPRODUCIBILITY AND ARTIFACT AVAILABILITY

The complete article-facing project snapshot is the public Radixor repository at commit 0c3b13f, and the frozen split protocol identifier is `radixor-generalization-v1`. The authoritative raw counters are stored in `docs/benchmarks/data/dictionary-generalization.csv`. Every raw row records the software/source identity, language, model ID, model version, compressed-model SHA-256, seed, requested coverage, selected and total row counts, overlap exclusions, and integer metric numerators and denominators [19].

The artifact-validation pipeline validates the complete $20 \times 10 \times 5$ matrix, selected-row cardinalities, count partitions, overlap arithmetic, model provenance, and the 100% boundary. Documentation is regenerated from checked-in counters and verification fails if the generated results or checksum are stale. The experiment source additionally verifies full-coverage output equivalence between the experimental builder and the released model loader for every parsed dictionary form before split evaluation proceeds.

The accompanying source package contains the derived CSV files used by the figures and tables, including the endpoint median/IQR summary and descriptive split ranges for the selected curves. These files are convenience extracts; the checked-in repository counters remain authoritative.

X. CONCLUSION

Across 20 UniMorph-derived language resources, a Radixor trie induced from a subset of dictionary families successfully applies training-derived patch-command behavior to many changed surfaces that were excluded from the exact training-surface set. The resource-macro unseen changed-exact value is 60.49% in the 10% coverage condition and 69.07% in the 90% condition; the corresponding across-resource medians are 63.72% and 70.44%. All 20 resource medians are higher at the 90% endpoint than at the 10% endpoint, with a median descriptive difference of 7.76 percentage points.

The principal result is not that one universal coverage curve characterizes stemming, nor that adding families has a quantified causal effect, nor that the model has demonstrated out-of-domain or contextual linguistic competence. It is that, under explicit family holdout and exact-surface leakage control, a lexicon-trained patch-command model exhibits measurable within-resource transfer beyond direct training membership across 20 languages. Because all default inputs are UniMorph-derived, this evidence is predominantly inflectional and resource-framework-specific.

This bounded conclusion is sufficient to distinguish the measured system from pure exact lookup while leaving stronger questions—fixed-test learning effects, independent-resource generalization, contextual ambiguity resolution, downstream retrieval effectiveness, command-inventory saturation, derivational morphology, and comparative stemming quality—to experiments designed specifically for those purposes.

REFERENCES

- [1] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge, U.K.: Cambridge University Press, 2008.

- [2] J. Singh and V. Gupta, "A systematic review of text stemming techniques," *Artificial Intelligence Review*, vol. 48, no. 2, pp. 157–217, 2017.
- [3] J. B. Lovins, "Development of a stemming algorithm," *Mechanical Translation and Computational Linguistics*, vol. 11, no. 1–2, pp. 22–31, 1968.
- [4] M. F. Porter, "An algorithm for suffix stripping," *Program*, vol. 14, no. 3, pp. 130–137, 1980.
- [5] R. Krovetz, "Viewing morphology as an inference process," *Artificial Intelligence*, vol. 118, no. 1–2, pp. 277–294, 2000.
- [6] C. D. Paice, "An evaluation method for stemming algorithms," in *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1994, pp. 42–50.
- [7] D. A. Hull, "Stemming algorithms: A case study for detailed evaluation," *Journal of the American Society for Information Science*, vol. 47, no. 1, pp. 70–84, 1996.
- [8] L. Weissweiler and A. Fraser, "Developing a stemmer for german based on a comparative analysis of publicly available stemmers," in *Language Technologies for the Challenges of the Digital Age*, ser. Lecture Notes in Artificial Intelligence. Springer, 2018, vol. 10713, pp. 81–94.
- [9] P. Majumder, M. Mitra, S. K. Parui, G. Kole, P. Mitra, and K. Datta, "YASS: Yet another suffix stripper," *ACM Transactions on Information Systems*, vol. 25, no. 4, pp. 18:1–18:20, 2007.
- [10] T. Brychcín and M. Konopík, "HPS: High precision stemmer," *Information Processing & Management*, vol. 51, no. 1, pp. 68–91, 2015.
- [11] J. Kanis and L. Müller, "Automatic lemmatizer construction with focus on OOV words lemmatization," in *Text, Speech and Dialogue*, ser. Lecture Notes in Artificial Intelligence, vol. 3658. Springer, 2005, pp. 132–139.
- [12] B. Jongejan and H. Dalianis, "Automatic training of lemmatization rules that handle morphological changes in pre-, in- and suffixes alike," in *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*. Suntec, Singapore: Association for Computational Linguistics, 2009, pp. 145–153. [Online]. Available: <https://aclanthology.org/P09-1017/>
- [13] T. Bergmanis and S. Goldwater, "Context sensitive neural lemmatization with Lematus," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. New Orleans, Louisiana: Association for Computational Linguistics, 2018, pp. 1391–1400. [Online]. Available: <https://aclanthology.org/N18-1126/>
- [14] —, "Data augmentation for context-sensitive neural lemmatization using inflection tables and raw text," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, 2019, pp. 4119–4128. [Online]. Available: <https://aclanthology.org/N19-1418/>
- [15] L. Galambos, "Lemmatizer for document information retrieval systems in JAVA," in *SOFSEM 2001: Theory and Practice of Informatics*, ser. Lecture Notes in Computer Science, vol. 2234. Springer, 2001, pp. 243–252.
- [16] —, "Semi-automatic stemmer evaluation," in *Intelligent Information Processing and Web Mining*. Springer, 2004, pp. 209–218.
- [17] A. Bialecki, "Stempel – algorithmic stemmer for the polish language," Technical/software documentation, preserved in the Apache Lucene Stempel documentation, 2004, testing section documents shuffled inflection sets, nested training samples, and a disjoint unseen-word test set. [Online]. Available: https://lucene.apache.org/core/9_12_3/analysis/stempel/index.html
- [18] K. Batsuren, O. Goldman, S. Khalifa, N. Habash, W. Kieraś, G. Bella, B. Leonard, G. Nicolai, K. Gorman *et al.*, "UniMorph 4.0: Universal morphology," in *Proceedings of the 13th Conference on Language Resources and Evaluation (LREC 2022)*. Marseille, France: European Language Resources Association, 2022, pp. 840–855. [Online]. Available: <https://aclanthology.org/2022.lrec-1.89/>
- [19] L. Galambos, "Radixor source and generalization experiment snapshot," GitHub repository, commit 0c3b13f485a9ad0b460c0931e4497ea95bed66a1, 2026, generalization protocol: radixor-generalization-v1. Accessed Aug. 26, 2026. [Online]. Available: <https://github.com/leogalambos/Radixor/tree/0c3b13f485a9ad0b460c0931e4497ea95bed66a1>