Journal of Kyzylorda Scholarly Review

P-ISSN:3006-0265 | E-ISSN:3007-4681. Published by the Open University of Kyzylorda
Copyright © Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC-BY).
https://bulletin.ouk.kz/index.php/bulletin/index




# DYNAMIC MULTI-SENSE SEMANTIC GRAPH LEARNING FOR HASHTAG UNDERSTANDING AND SEMANTIC DRIFT DETECTION IN SOCIAL MEDIA

**Suhad Malallah Kadhem**[*1]
[1]Collage of Computer Science, University of Technology- Iraq, Baghdad, Iraq
[*]Correspondance: suhad.m.kadhem@uotechnology.edu.iq

**Article History:**
Received: 30th Sept 2026; Revised: 20th Oct 2026; Accepted: 10th Nov 2026; Published: 25th Nov 2026

**Abstract:** A hashtag can be used in a variety of ways, and even in different ways on the same day. Therefore, it is possible that the changes in its usage are concealed when its occurrences counted or being represented by one average vector. This study proposes a framework for representing English language hashtags through contextual sense nodes, called DMS-HGraph, which can maintain their identity across successive time windows. The technique uses density-based clustering, target-aware contextual embeddings, a weighted graph construction, and threshold temporal alignment, also it distinguishes movement in persistent senses, redistribution of sense prevalence and the disappearance or appearance of the senses. These mechanisms are isolated in a numerical, controlled evaluation in five contexts: a stable control, a shift in context with no frequency change. These scores show the desired effect of the suggested measures using known inputs and do not establish the accuracy of the detection on real social media. For subsequent empirical testing, a chronological evaluation protocol, annotation strategy, and baseline comparisons are given. The framework is designed for interpretable analysis of evolving discussions, and not for the purpose of inferring the beliefs or intentions of individual users. An additional controlled text level pilot uses a total of 720 posts generated in English, for testing clustering and temporal alignment with known categories and it provides insight on the behavior of the method under separable vocabularies but it is not used to show performance on authentic social media data.

**Keywords**: Contextual embeddings, hashtag analysis, dynamic graphs, semantic drift, social media mining, sense induction.

## Introduction

Hashtags are a compact way of organizing online conversation, they are short and easy to retrieve but can also be ambiguous – #Java can refer to a programming language, or coffee or an island. A sudden surging of the programming uses of the tag do not necessarily signify a new meaning. On the contrary, a new use appears while the quantity of postings stays the same. All these cases call for a representation that not only preserve their aggregate frequency of the contexts occurred, but also their distribution. Changing hashtag communities utilizing temporal co occurrence networks was studied by[1], while the combined temporal and semantic information with regard to event storylines were studied by[2]. Through these studies, it is determined that graph-based hashtag analysis and temporal hashtag analysis are existing research directions. The question of interest here is

more limited: Does an explicitly sense-resolved graph distinguish various types of contextual evolution as well as provide evidence supporting each? This proposal treats the contextual clusters as candidate usage senses not automatically as dictionary meanings.

## 1. Related Work

### 1.1 Contextual meaning and semantic change

The SemEval lexical semantic change benchmark[3] set a supervised evaluation reference for the unsupervised change scores. Kutuzov[4] warn that the contextual embeddings can conflate shifts in the lexical meanings with changes in contextual variation. This is pertinent to hashtags since the discussion of a new product may be a topic change within the existing referent. Sentence-BERT[5] offers efficient similarity representations and BER Tweet[6] is pretrained on English tweets. XL-LEXEME[7] explicitly marks the target expression and provides a suitable alternative to post-level pooling. Periti and Tahmasebi[8] compare contextual methods and test them under common conditions and encourage testing of several encoders and measures instead of selecting single architecture by assumption. Word-in-Context[9] offers a helpful formulation regarding the question of whether two occurrences share a meaning. DWUG[10] generalizes this to human-annotated use graphs, and More DWUGs[11] explores annotation robustness. Ma[12] suggest graph-based clustering to detect the gain and loss of word senses. Kutuzov[13] add cluster definitions to usage graphs. Put together, these studies offer close precedents to the sense-level representations and a 'hashtag graph' alone is nothing new.

### 1.2 Social media and temporal graph modeling

TimeLMs[14] addresses the temporal domain variation of Twitter language and TweetEval[15] standardizes a number of social media classification tasks. Both fail to provide a labeled benchmark for hashtag sense transitions. Longa[16] make a distinction between snapshot and event driven graph settings in their temporal graph survey. A snapshot design is chosen here since a sense distribution and its uncertainty is estimated in a fixed window.

### 1.3 Competing drift measures

Important baselines are the average embedding and pairwise distance methods[17]. Asgari[18] do a comparison of embedding-based profiles and Martinc[19] combine the contextual clustering and static embeddings. Karnysheva and Schwarz[20] have already used clustering and Jensen–Shannon comparisons. More recently, there are interpretable or instance level alternatives in the form of lexical replacement analysis[21] and unbalanced optimal transport[22]. Goworek and Dubossarsky[23] explore minimum-distance metrics while Ling and Komachi[24] explore interpretable sense distributions. The evaluation of the proposed contribution has to take into account these recent methods.

### 1.4 Critical synthesis and operational definitions

The unit of analysis is an occurrence of the hashtag, not the hashtag string or the author. A sense is empirically supported cluster of usages that are interpreted at a predefined granularity level. For instance, two different releases of Java software usually refer to the same referent, but Java as an island is different from Java as a programming language. It is important to differentiate between the two: a model designed to detect topical variation can be useful even if it is not a lexical sense disambiguator. So, the annotation protocol distinguishes between referent, topic and uncertainty. The close alternatives are recent graph clustering work optimal transport method, and interpretable distributional analysis but the proposed contribution is not a claim that temporal distributions or sense clustering are new, yet an integration for hashtag graphs.

## 2. Research Gap and Questions

Proposed contribution can be seen, in the context of a hashtag specific integration of within-window sense induction, cross window identity alignment as well as graph linking various hashtags at sense level. The existing lexical change approaches offer good alternatives to the individual lexical drift components. Thus, the research needs to be done in order to understand if the integration provides measurable value instead of claiming the clustering, graph construction or distributional change is individually new.

| Question | Testable hypothesis |
|---|---|
| RQ1: Sense induction | Target-aware representations improve agreement with annotated hashtag usages over post-level pooling. |
| RQ2: Temporal alignment | Multi-match alignment improves event detection over one-to-one matching. |
| RQ3: Graph contribution | Sense-level neighbor evidence improves drift detection over the same model without graph edges. |
| RQ4: Interpretability | Separate drift components help reviewers identify the source of a change. |

## 3. Proposed Framework

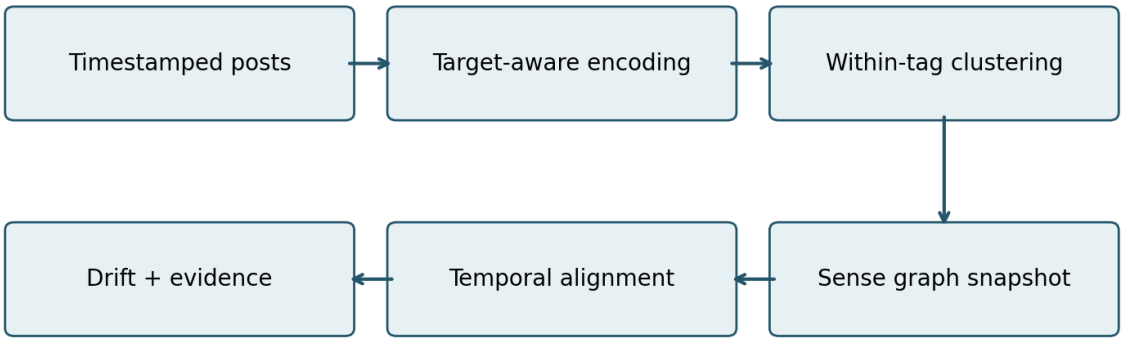



**Figure 1.** Proposed processing pipeline. The previous graph is retained for temporal comparison.

Input are timestamped English posts. Pre-processing steps involved removing out duplicate records, replacing user mentions and URLs by using placeholders, and retaining only the target hashtag and surrounding context. Each occurrence will be mapped to a vector using the encoder. Clustering groups the usages of the same hashtag in a time window. Every group is turned into a node of a weighted graph. The temporal alignment links the nodes from adjacent windows, with a final report reports component level drift and representative evidence. The pipeline is causal, meaning that a prediction for window t will only use observations up to the end of window t.

## Materials and Methods

### 1. Occurrence representation and sense induction

So we can assume a post is p = (id, x, H, t), where x is the text, H is the set of hashtags and t is the timestamp. Utilize disjoint weekly windows, daily/monthly options for sensitivity analysis. A frozen encoder maps marked context of a target hashtag h to a unit vector z =

fθ(x,h)/‖fθ(x,h)‖. If a single encoder is used for all the windows, then there is no need to compare incompatible vector spaces.

$$C(h,t) = \text{HDBSCAN}(\{z_1, \ldots, z_n\}) \quad (1)$$

Then a candidate sense k has centroid $\mu_k$ equal to normalized mean regarding its assigned vectors as well as prevalence $q_k = n_k / N$, where N is the number of non-noise occurrences. Report noise coverage in a separate way. A minimum number of occurrences per tag/window (initially 30) and cluster (initially 5) are required and abstain if a window is too sparse. These are initial values, NOT optimized settings.

**2. Weighted semantic graph**

$$G_t = (V_t, E_t, X_t, W_t) \quad (2)$$

A node v = (h,k,t,μ,q,L) stores the tag, cluster identifier, time, centroid, prevalence and representative lexical terms L, and edges are connect the senses of different hashtags. For nodes i and j let $P_i$ and $P_j$ be their assigned post sets; let $a_i$ and $a_j$ be activity vectors over sub-windows.

$$w_{ij} = \alpha(1 + \cos(\mu_i,\mu_j))/2 + \beta\, J(P_i,P_j) + \gamma \cos(a_i,a_j) \quad (3)$$

J represent the intersection size divided by union size (zero for empty union). The activity vectors are nonnegative and zero similarity is provided in the case of zero norm. All of the coefficients are non-negative and sum to 1 and retain edges above τe and among top K neighbor per node; symmetrize by union. Initial setting is α=0.5, β=0.3, γ=0.2, K=10. Tune on the validation windows.

**3. Temporal alignment**

$$A_{ij} = \lambda(1 + \cos(\mu_i,\mu_j))/2 + (1 - \lambda)\, J(L_i,L_j) \quad (4)$$

Only compare sense of the same hashtag in adjacent windows and retain all matches with A ≥ τa; and allow one-to-many, and many-to-one events. A candidate birth is a node without predecessor and a candidate loss is a node without successor. Verify the events over second valid window. In the case when it's the window that's missing, abstain, don't declare loss. Many to many components are marked ambiguous, while split or a merge is a hypothesis concerning usage clusters, and can not be used to prove the existence of a linguistic change.

**4. Worked construction of a weighted semantic edge**

Let the technology sense of #Apple be posts {1,2,3,4} and #iPhone be posts {1,2,5}. Their intersection is of size 2, and their union of size 5, so their co-occurrence J = 0.4. Let the normalized centroid similarity be 0.90 and activity similarity be 0.80. With weights α = 0.5, β = 0.3 and γ = 0.2, the edge weight is 0.5(0.90) + 0.3(0.40) + 0.2(0.80) = 0.73. This edge is retained at a threshold value of 0.60. Unrelated fruit node with same temporal activity and no co-occurrence with semantic similarity 0.20 would get 0.26 and be rejected. The inputs are indicative and not actual social media relationships.

| Pair | Semantic | Co-occurrence | Activity | Weight | Decision |
|---|---|---|---|---|---|
| Apple-tech / iPhone | .90 | .40 | .80 | .73 | Retain |
| Apple-fruit / iPhone | .20 | 0 | .80 | .26 | Reject |

**5. Worked temporal alignment and graph update**

Take into account the two prior senses of (#Java), programming and coffee, and the three present senses, programming, coffee and island. Assume that their alignment scores are as indicated below. At τa = 0.80, the two diagonal matches persist and the island sense has no predecessor. Had there been two accepted links to distinct current programming contexts in

the programming row, then it would be marked by the system as as a split candidate and not as a confirmed lexical innovation. For a human reviewer to look through some representative posts and see if the event is an actual sense distinction.

| Previous / current | Programming | Coffee | Island |
|---|---|---|---|
| Programming | 0.94 | 0.32 | 0.41 |
| Coffee | 0.25 | 0.91 | 0.36 |

## 6. Drift Estimation and Graph Evidence

Let M be the accepted correspondence set: For each pair apply a normalized weight proportional to the min ($q_i$, $q_j$); in the case of many-to-many matching, this is a descriptive weighting rather than a conserved transport plan. The contextual component is undefined, while the combined score is withheld or labeled separately as structural only alert, if M is empty.

$$D\square = \Sigma(i,j)\in M\ \omega_{ij}\ (1 - \cos(\mu_i,\mu_j))/2 \qquad (5)$$

Find a common inventory of connected components of bipartite temporal alignment graph, where unmatched nodes are singleton components. Sum all the window sense masses in each component to get distributions P and Q. This prevents arbitrarily allocating mass in the split/merge, but a redistribution in a component is deliberately not measured.

$$D_p = ½\ KL_2(P||M_0) + ½\ KL_2(Q||M_0),\ M_0 = (P+Q)/2 \qquad (6)$$

Apply base-2 logarithms, assuming that the zero-mass terms are equal to zero. If B is the probability mass of unmatched current nodes and L that of unmatched previous nodes.

$$D_s = (B + L)/2;\ D = \eta_1 D\square + \eta_2 D_p + \eta_3 D_s \qquad (7)$$

The weights sum to 1. The initial weights are (0.4, 0.4, 0.2). The three scores are still visible, as birth and loss rates could also have impact distributional drift; the aggregate is heuristic, not independent decomposition of probability. Any split/merges are reported in a separate way.

### 6.1 Graph-aware extension

The graph contributes additional measurable feature: for persistent one-to-one matches: compare the weighted neighbor distributions over aligned neighbor identities. Take Jensen–Shannon divergence of each pair and then average by matched prevalence. Call this Dg, compare a graph-free score D with graph-aware score $(1-\rho)D+\rho Dg$ where ρ is selected on validation data. When there are no neighbors for either of the two nodes, abstain, and such extensions makes the graph contribution testable: only D, not Dg is evaluated by the controlled a numerical demonstration below.

### 6.2 Proposed algorithm: incremental DMS-HGraph update

The proposed snapshot based procedure is formalized in Algorithm 1. It takes as input timestamped English posts, frozen target-aware encoder, clustering configuration and a set of validation-selected parameters. Its products include: Sequence of sense-level graphs, Temporal correspondence records, component drift scores and supporting evidence. The algorithm is applied to one time window at a time and doesn't need access to the future test windows. In Section 7, the simpler configuration described is used for the controlled pilot, implementing only the clustering, alignment and drift core.

**Algorithm 1. Incremental construction and semantic drift estimation (DMS-HGraph)**

Input: posts D; window length Δ; encoder fθ; clustering parameters; edge threshold τe; alignment threshold τa; weights α, β, γ, λ, η and ρ; minimum support nmin; neighbor limit K.

Output: snapshots G; temporal matches M; event records E; drift records R with evidence.

01 Initialize G, M, E, R and per-hashtag history as empty.
02 Normalize posts; retain identifiers, timestamps and target hashtags.
03 Partition posts chronologically into disjoint windows $D_1$, …, $D_t$.
04 for each window t do
05 Initialize current sense inventory $V_t$ and coverage records.
06 for each hashtag h observed in the window do
07 Collect its occurrences; if support < nmin, mark unknown and continue.
08 Encode marked contexts using fθ; L2-normalize the vectors.
09 Cluster vectors with HDBSCAN; retain eligible clusters and record noise.
10 For each cluster, compute centroid, prevalence, terms and evidence IDs.
11 Add the resulting sense nodes to $V_t$.
12 end for
13 Compute Eq. (3) for candidate cross-hashtag node pairs.
14 Keep thresholded top-K neighbors; symmetrize to construct $G_t$.
15 for each hashtag with adequate coverage in adjacent windows do
16 Compute the alignment matrix using Eq. (4).
17 Retain every pair with A ≥ τa; store the bipartite match graph.
18 Label unmatched nodes as candidate births or losses.
19 Label one-to-many and many-to-one matches as splits and merges;
20 flag many-to-many components as ambiguous.
21 Build the common inventory from alignment connected components.
22 Sum prevalence in each component to obtain P and Q.
23 Calculate distributional and structural drift by Eqs. (6)–(7).
24 If matches exist, calculate contextual drift by Eq. (5) and score D;
25 otherwise record structural-only evidence and withhold D.
26 If eligible aligned neighbors exist, calculate Dg and graph-aware score;
27 otherwise retain D and mark graph evidence unavailable.
28 Attach component scores, matches, evidence IDs and coverage to R.
29 end for
30 Confirm pending birth/loss events when the next valid adjacent window arrives;
31 retain missing-window cases as unknown, not as confirmed losses.
32 Save $G_t$, temporal matches and event records; advance the history.
33 end for
34 Return G, M, E, R.

**6.3 Detailed description of Algorithm 1**

Lines 1-12 do the occurrence to sense transformation. The target marker will cause the encoder to be sensitive to the particular hashtag instead of equally treating each word in a post. Within each hashtag, separate clustering prevents unrelated hashtags from sharing sense identifier. The centroid summarizes the cluster, while the prevalence corresponds to the percentage of assigned occurrences. Insufficient coverage as well as noise are preserved

as uncertainty not converted silently into new meanings. The semantic graph is constructed in lines 13–14. However, for a high edge weight, there is a requirement for evidence from the contextual similarity, shared posts or of synchronized activity. The control graph density is limited by threshold and neighbor. For instance, in Section 5.4, Apple technology–iPhone relationship is given a weight of 0.73, while fruit–iPhone is given 0.26, which is too low to be accepted with a threshold of 0.60. The graph thus contains more information than an unweighted co-occurrence network. The Temporal Identity is set in lines 15-22. The algorithm is designed to maintain several accepted correspondences, not imposing one to one assignment. For example, new island cluster for #Java becomes candidate birth in the case when neither previous programming cluster or previous coffee cluster reach the alignment threshold. Topology of temporal correspondence graph represents a split or merge. Connected components are then used to specify an aligned inventory to compare prevalence without arbitrary relabeling. Lines 23–28 quantify as well as explain change. Contextual drift can capture persistent centroids movement, distributional drift can capture redistributions of prevalence, and unmatched probability mass can be captured by structural drift. The optional graph term is measuring the changes in aligned neighborhoods. By returning back the component scores and the original evidence identifier, it becomes possible to differentiate between the reasons of the identical same aggregate score. Lines 29-34 continue to show the temporal state. A cluster being missing in one window cannot indicate that the candidate disappearance is confirmed. Alerts may be delayed by a window since enough coverage and a subsequent valid observation are required for confirmation. Evens are provisional for the last available window. The algorithm records snapshots instead of overwriting the previous snapshots and allowing reproducible chronological analysis.

**Results and Discussion**

Here we have two evaluations of DMS-HGraph that are complementary. The first isolates the numerical behaviour of the proposed drift measures under known changes in sense prevalence and centroid location. The second takes a different approach and tries to find out whether such changes can be recovered from the generated English posts with a simplified text-processing implementation. The evaluation aims to address three questions: first, do the component scores respond to the type of intended change? second, are the reference categories recovered in the sense induction stage? third, is the temporal description conveys information absent from frequency and single-centroid baselines? These experiments are not tests using data from actual social media, but rather a controlled demonstration of the experiments.

**1. Numerical verification of the drift measures**

Consider 1000 hypothetical occurrences in each one of the windows. Technology and fruit are recognized contexts, a new context related to AI is added as a separate candidate for demonstration. In practice an AI product discussion could be a subtopic of technology and need to be validated by annotation.

| Candidate context | Window 1 | Window 2 |
|---|---|---|
| Technology | 800 | 500 |
| Fruit | 200 | 200 |
| AI-related | 0 | 300 |

The two persistent centroids are not moving and AI is assumed to be unmatched, in the case of the arithmetic example. The aligned distributions are P=(0.8,0.2,0) and Q=(0.5,0.2,0.3). Their mean is (0.65,0.2,0.15). The divergence is calculated as the square of scipy. spatial. distance. Jensen Shannon (P, Q, base=2) and the square is important as library returns distance and not divergence. The distributional drift is 0.1752; structural drift: (0.30+0)/2 = 0.1500, contextual drift: 0, combined score is 0.1001. The total posts will not reflect a change thus a pure volume measure does not report a change.

**2. Response to distinct semantic-change scenarios**

| Scenario | Context | Distribution | Structure | Combined |
|---|---|---|---|---|
| Stable | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| Redistribution | 0.0000 | 0.1912 | 0.0000 | 0.0765 |
| Birth | 0.0000 | 0.1752 | 0.1500 | 0.1001 |
| Loss | 0.0000 | 0.1752 | 0.1500 | 0.1001 |
| Context rotation | 0.0536 | 0.0000 | 0.0000 | 0.0214 |

The redistribution scenario moves prevalence - from (0.8,0.2,0) to (0.3,0.7,0) - without affecting the centroids. The loss scenario reverse birth example, and the context-rotation scenario places the first persistent centroid at 30 degree, and keeps its prevalence at 0.8 whilst the second remains fixed. All correspondences are stipulated and nothing is changed by the stable control. The five scenarios form the basis that defines the behavior of the measure in the deliberately isolated conditions. For the stable case, every component is equal to zero as desired. Redistribution generates a positive distributional score of 0.1912, without activating either the contextual or structural component. Both the birth and the loss yield aggregate score (0.1001), as the measure of divergence and unmatched-mass are symmetric, so the direction should be interpreted from the event record and not from the scalar score. A 30-degree rotation of persistent centroid, on the other hand, causes a contextual drift of only 0.0536 and leaves the prevalence distribution unchanged. These results confirm that the proposed representation is able to express qualitatively different changes without changing the posting volume. Not statistically calibrated detection threshold is established.

**3. Experimental design and implementation of the text-level pilot**

A small end-to-end pilot was run on generated English texts to complement the analytical scenarios. It assesses the ability of the clustering and temporal alignment mechanism to recover the known contextual categories given deliberately separable vocabulary. It does not assess BERTweet, HDBSCAN, learned temporal graph network, or the actual social media language. The pilot, based on TF-IDF and average-linkage agglomerative clustering, is a very simple realization of the proposed modular pipeline. The reference labeling is assigned by the generation, rather than human annotators. The corpus contains 720 records: three hashtags (#Apple, #Python, #Java), four windows, and 60 occurrences per hashtag per window. Each post draws four distinct terms from a six-word sense-specific vocabulary and one common background word. The three sense inventories are technology/fruit/music, programming/reptile/comedy and programming/coffee/island, respectively. The fixed random seed is 42. The prevalence schedules are identical across hashtags to isolate the effect of vocabulary and sampling.

Journal of Kyzylorda Scholarly Review
P-ISSN:3006-0265 | E-ISSN:3007-4681
https://bulletin.ouk.kz/index.php/bulletin/index

| Window | Sense 1 | Sense 2 | Sense 3 | Designed transition |
|---|---|---|---|---|
| 1 | 48 | 12 | 0 | Initial |
| 2 | 30 | 30 | 0 | Redistribution |
| 3 | 24 | 18 | 18 | Birth |
| 4 | 0 | 24 | 36 | Loss |

TF-IDF is fitted jointly to the generated corpus for this descriptive pilot; this is not a prospective evaluation because future vocabulary is available. Vectors are L2-normalized. Average-linkage clustering uses cosine distance and a cut of 0.65 without supplying the true number of clusters. Each centroid is normalized after averaging. For the pilot, alignment uses normalized cosine alone ($\lambda = 1$), with threshold 0.80. The connected components of the bipartite alignment graph provide the shared inventory for Jensen–Shannon divergence. The structural score records unmatched mass; contextual drift averages matched centroid distances, weighted by minimum prevalence. Graph-neighborhood evidence is not included in the pilot score.

## 4. Performance of contextual sense induction

| Hashtag | Window | True senses | Detected | ARI | NMI |
|---|---|---|---|---|---|
| Apple | 1 | 2 | 2 | 1.000 | 1.000 |
| Apple | 2 | 2 | 2 | 1.000 | 1.000 |
| Apple | 3 | 3 | 3 | 1.000 | 1.000 |
| Apple | 4 | 2 | 2 | 1.000 | 1.000 |
| Python | 1 | 2 | 2 | 1.000 | 1.000 |
| Python | 2 | 2 | 2 | 1.000 | 1.000 |
| Python | 3 | 3 | 3 | 1.000 | 1.000 |
| Python | 4 | 2 | 2 | 1.000 | 1.000 |
| Java | 1 | 2 | 2 | 1.000 | 1.000 |
| Java | 2 | 2 | 2 | 1.000 | 1.000 |
| Java | 3 | 3 | 3 | 1.000 | 1.000 |
| Java | 4 | 2 | 2 | 1.000 | 1.000 |

In all 12 hashtag-window combinations, the clustering results are on exact agreement with the generator assigned reference categories. Adjusted Rand index (ARI) as well as normalized mutual information (NMI) attain 1.000 in each one of the combinations and give micro average (1.000) for the two measures. The number of clusters detected corresponds to the desired sequence 2, 2, 3 and 2 for the hashtags. Of note, the true number of clusters was not provided to the agglomerative clustering procedure. This means that the selected distance threshold was enough for recovering deliberately separated groups in corpus. The generated vocabularies are of limited overlap, and the same corpus was used to fit the TF-IDF, so it is not to be interpreted as evidence of perfect disambiguation on real posts. The experiment does not test the generalization ability of the pilot, but only verifies the behavior of the pilot in favorable conditions.

## 5. Temporal alignment and semantic drift results

| Tag | Transition | Matches | Birth | Loss | D_C | D_P | D_S | D |
|---|---|---|---|---|---|---|---|---|
| Apple | 1–2 | 2 | 0 | 0 | 0.0044 | 0.0731 | 0.0000 | 0.0310 |
| Apple | 2–3 | 2 | 1 | 0 | 0.0105 | 0.1722 | 0.1500 | 0.1031 |

Journal of Kyzylorda Scholarly Review
P-ISSN:3006-0265 | E-ISSN:3007-4681
https://bulletin.ouk.kz/index.php/bulletin/index

| Tag | Transition | Matches | Birth | Loss | D_C | D_P | D_S | D |
|---|---|---|---|---|---|---|---|---|
| Apple | 3–4 | 2 | 0 | 1 | 0.0146 | 0.2419 | 0.2000 | 0.1426 |
| Python | 1–2 | 2 | 0 | 0 | 0.0072 | 0.0731 | 0.0000 | 0.0321 |
| Python | 2–3 | 2 | 1 | 0 | 0.0107 | 0.1722 | 0.1500 | 0.1032 |
| Python | 3–4 | 2 | 0 | 1 | 0.0214 | 0.2419 | 0.2000 | 0.1454 |
| Java | 1–2 | 2 | 0 | 0 | 0.0141 | 0.0731 | 0.0000 | 0.0349 |
| Java | 2–3 | 2 | 1 | 0 | 0.0089 | 0.1722 | 0.1500 | 0.1025 |
| Java | 3–4 | 2 | 0 | 1 | 0.0061 | 0.2419 | 0.2000 | 0.1392 |

Transitions indicate that the consistent ordering of aggregate drift is maintained for the 9 transitions across the three hashtags. In the first transition, the two senses persist and their relative proportions shift from 0.8/0.2 to 0.5/0.5. The distributional divergence is in all three tags 0.0731 and the structural component is 0. The range of aggregate scores is 0.0310 to 0.0349; differences in scores are not indicative of the designed prevalence schedule, but of estimated centroid scores. A third contextual category accounts for 30% of the posts in the second transition. The structural component goes up to 0.1500 and the aggregate scores go up to 0.1025–0.1032. The last transition removes a category with previous prevalence 0.40 and gives a structural drift of 0.2000 and an aggregate score of 0.1392-0.1454. The loss transition then gives the largest score in this particular controlled schedule and this ordering is not a universal ranking over the type of event: it relies on the affected probability mass as well as the chosen weights.

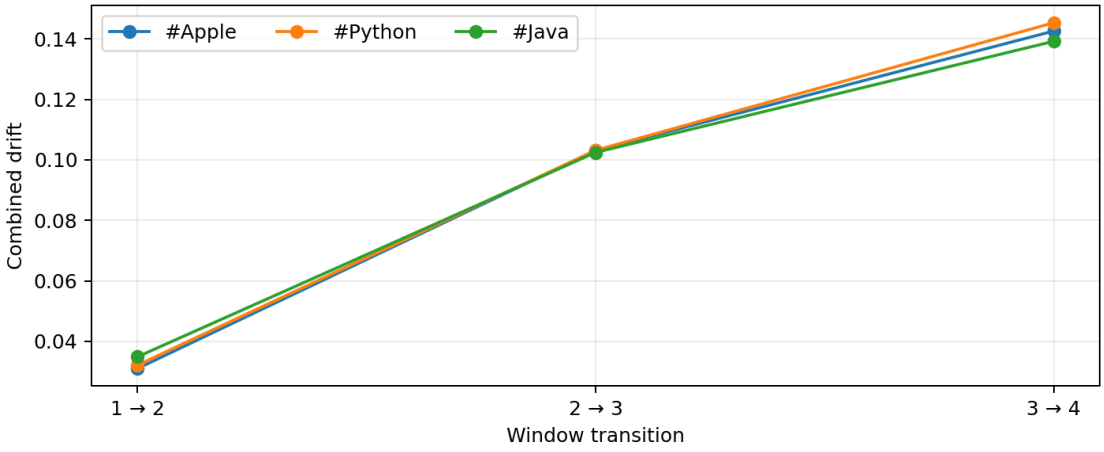



**Figure 2.** Computed drift scores on the generated English text corpus.

The mean combined drift is 0.0327 along the redistribution transition, 0.1029 along the birth transition and 0.1424 along the loss transition for all three tags. The contextual component is rather modest (0.0044-0.0214), keeping with the random sampling from fixed sense-specific vocabularies, than intentional centroid movement. The counts in temporal results table recover two accepted matches in each transition, while no candidates' births or losses in first transition, one candidate birth in second and one candidate loss in third for each one of the hashtags. Important: These counts should be interpreted only in the descriptive sense; for precision and recall at the event level, one would need independently verified correspondence labels and more negative examples. Specifically, the pilot doesn't use the two window confirmation policy of Algorithm 1.

## 6. Comparison with frequency and single-centroid baselines

| Hashtag | Transition | Volume change | Centroid distance | Proposed score |
|---|---|---|---|---|
| Apple | 1–2 | 0.0000 | 0.0696 | 0.0310 |
| Apple | 2–3 | 0.0000 | 0.0769 | 0.1031 |
| Apple | 3–4 | 0.0000 | 0.1406 | 0.1426 |
| Python | 1–2 | 0.0000 | 0.0717 | 0.0321 |
| Python | 2–3 | 0.0000 | 0.0782 | 0.1032 |
| Python | 3–4 | 0.0000 | 0.1482 | 0.1454 |
| Java | 1–2 | 0.0000 | 0.0790 | 0.0349 |
| Java | 2–3 | 0.0000 | 0.0770 | 0.1025 |
| Java | 3–4 | 0.0000 | 0.1356 | 0.1392 |

The frequency based monitoring returns 0 change regarding each one of the transitions since the number of occurrences is fixed (60) per window and hashtag. This is a direct result of the experimental design and is a limitation of volume-only monitoring. In all nine transitions, the single-centroid baseline returns positive distances, between 0.0696 and 0.1482. This means that it can be used to recognize any aggregate contextual movement without altering the posting volume. The single distance is not indicative however of movement due to the emergence of a new category or due to a disappearing category or redistribution in persistent categories. DMS-HGraph distinguishes between by using temporal correspondence records and component scores. The centroid distances of redistribution and birth are 0.0696 and 0.0769, and the corresponding proposed scores are 0.0310 and 0.1031 for #Apple, respectively. This distinction in interpretation between the two transitions is due to the explicit structural component. Because the measures are defined, and scaled differently, it is not possible to make judgements of improved detection accuracy based on these differences in numbers. No statistical superiority claims are made.

## 7. Interpretation, failure modes and threats to validity

The results of the controlled experiments support the internal consistency regarding the suggested drift formulation, and show its interpretability when the sense categories are known or can be easily separated. There are several limitations, however, that limit the conclusions. First, the corpus is generated from disjoint vocabularies and lack lexical overlap, figurative language, sarcasm and incomplete contexts found in social media. Second, future vocabulary information is added when fitting TF-IDF on the entire corpus, so it is a retrospective pilot evaluation, not a prospective evaluation. Thirdly, small or noisy collection of clusters could disappear due to chance even when its underlying sense is active. The minimum support requirements, coverage reporting and confirmation across adjacent windows is therefore required in a real stream. Fourth, the observations are not independent evidence regarding performance over various temporal patterns as indicated through identical prevalence schedule in all the three tags. Experiments are not for the complete graph-aware extension. For the text-level pilot, only TF-IDF, clustering, temporal alignment and the three-component drift measure are used, as shown in Section 5 to construct weighted cross-hashtag edges. Here, neither contextual transformer embeddings nor the graph-neighborhood learning is assessed. Thus, the results demonstrate only feasibility regarding core procedure under the controlled conditions, yet are not definitive in terms of

the added value of graph edges, performance on natural language or superiority over recent semantic-change procedures. Section 8 outlines the chronological benchmark as well as ablation experiments that can be performed to address such questions.

## 8. Empirical Evaluation Protocol

### 8.1 Data and annotation

If allowed and available, use the hashtag stream of and separately, collect English posts from ambiguous hashtags, namely #Apple, #Python, #Java, #Jaguar, #Mercury, etc. The 100,000 posts is a planning target based on access and coverage. Retain the timestamps as well as pseudonymous record identifiers, don't redistribute the restricted text. Deduplicate the near identical posts and report sampling policy, missing windows and language filtering. A stratified sample of occurrences over time should be labeled by two independent annotators, separating referent changes, topic variations, from ambiguous contexts and noise. Apply pairwise same-sense judgments, as in and resolve disagreement. Annotate pairs of windows for temporal events without model scores shown. Report on agreement and number of adjudicated cases. TweetEval and SemEval can offer auxiliary testing, but does not replace a hashtag-specific temporal benchmark.

### 8.2 Chronological split and baselines

Use the earliest 60% of windows for development, the next 20% for validation, while the final 20% for testing and keep the near duplicate posts in one split. Adjust encoder, clustering setting, alignment as well as drift levels prior to the test evaluation and consider adding a "held out hashtag" experiment to evaluate the generalization beyond the known tags.

| Baseline | Comparison purpose |
|---|---|
| Frequency change | Volume-only control |
| Temporal co-occurrence [1] | Hashtag-level graph |
| Prototype cosine / APD [17] | Embedding-only change |
| Clustering + JSD [19], [20] | Sense distribution without graph |
| Graph-based clustering [12] | Strong sense-aware baseline |
| Optimal transport [22] | Instance-level change |
| Minimum distance [23] | Recent contextual metric |
| DMS-HGraph with / without Dg | Isolate graph contribution |

### 8.3 Metrics and reporting

Report adjusted Rand index as well as B-cubed F1 for clustering and pairwise F1 for temporal alignment, macro-F1 per event class, average precision, also Spearman correlation for drift and assess detection latency, graph update time, peak memory and abstention coverage. Use paired bootstrap confidence intervals by hashtag and report sensitivity to window size, threshold and encoder choices and measure runtime on the same hardware, and embedding cost and graph update cost.

## 9. Discussion and Limitations

The numerical results demonstrate that the proposed scores can be able to separate a number of controlled changes while the effectiveness of such framework depends on an effective sense extraction. A cluster may be a stylistic pattern, an event or named entity, but not a linguistic sense. Throughout, the term candidate sense is used and high semantic drift score is a call to see the evidence; it is not a definitive statement on language change. The

temporal alignment rule is transparent but subject to centroid quality as well as lexical overlap, while a sparse sense may not be seen in a sample without disappearing from population. This risk can be lowered, however, with confirmation over successive windows and confidence intervals and abstention, while increasing latency. Comparing to unbalanced optimal transport is especially relevant, since transport based alignment could account for the unequal mass without specifically depending on thresholded graph matches. Graph aware extension should earn its complexity and a useful graph should enhance interpretation or detection and do more than just clustering and divergence. Semantic edges, temporal activity, co-occurrence and structural drift and target marking will each be removed in an ablation study. No graph variant utilizes the same embeddings and cluster assignments to ensure a fair test of the additional value of graph structure.

## Conclusions

DMS-HGraph offers methodological framework to track tracking multiple contextual usages regarding English hashtags through time and the suggested algorithm integrates the target-aware sense induction, weighted graph construction, temporal alignment and interpretable drift estimation. A total of five numerical scenarios is able to validate the desired behavior related to the component measures and a controlled pilot on 720 generated posts reveals the ability of the simplified implementation to recover deliberately separable contextual categories and their intended temporal evolution. The results are for methodological feasibility, not efficiency in real world. The next step is to test the entire graph-aware framework on separately annotated chronological social media benchmark as well as quantify its impact compared to effective lexical and graph-based baselines.

## References

[1] C. M. Garcia, A. de Souza Britto Jr., and J. P. Barddal, "Temporal analysis of drifting hashtags in textual data streams: A graph-based application," *Expert Systems with Applications*, vol. 257, Art. no. 125007, 2024, doi: 10.1016/j.eswa.2024.125007.

[2] J. Song, Y. He, X. Zhang, N. Xu, and G. Fu, "Harnessing the power of hashtags: Temporal pattern mining and storyline construction for event evolution on social media," *PLOS ONE*, vol. 20, no. 8, Art. no. e0327596, 2025, doi: 10.1371/journal.pone.0327596.

[3] D. Schlechtweg, B. McGillivray, S. Hengchen, H. Dubossarsky, and N. Tahmasebi, "SemEval-2020 Task 1: Unsupervised lexical semantic change detection," in *Proc. SemEval*, 2020, doi: 10.18653/v1/2020.semeval-1.1.

[4] A. Kutuzov, E. Velldal, and L. Øvrelid, "Contextualized embeddings for semantic change detection: Lessons learned," *Northern European Journal of Language Technology*, vol. 8, 2022, doi: 10.3384/nejlt.2000-1533.2022.3478.

[5] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT-networks," in *Proc. EMNLP-IJCNLP*, 2019, pp. 3982–3992, doi: 10.18653/v1/D19-1410.

[6] D. Q. Nguyen, T. Vu, and A. T. Nguyen, "BERTweet: A pre-trained language model for English tweets," in *Proc. EMNLP System Demonstrations*, 2020, pp. 9–14, doi: 10.18653/v1/2020.emnlp-demos.2.

[7] P. Cassotti, L. Siciliani, M. DeGemmis, G. Semeraro, and P. Basile, "XL-LEXEME: WiC pretrained model for cross-lingual lexical semantic change," in *Proc. ACL*, 2023, doi: 10.18653/v1/2023.acl-short.135.

[8] F. Periti and N. Tahmasebi, "A systematic comparison of contextualized word embeddings for lexical semantic change," in *Proc. NAACL*, 2024, pp. 4262–4282, doi: 10.18653/v1/2024.naacl-long.240.

[9] M. T. Pilehvar and J. Camacho-Collados, "WiC: The word-in-context dataset for evaluating context-sensitive meaning representations," in *Proc. NAACL*, 2019, pp. 1267–1273, doi: 10.18653/v1/N19-1128.
[10] D. Schlechtweg, N. Tahmasebi, S. Hengchen, H. Dubossarsky, and B. McGillivray, "DWUG: A large resource of diachronic word usage graphs in four languages," in *Proc. EMNLP*, 2021, pp. 7079–7091, doi: 10.18653/v1/2021.emnlp-main.567.
[11] D. Schlechtweg et al., "More DWUGs: Extending and evaluating word usage graph datasets in multiple languages," in *Proc. EMNLP*, 2024, pp. 14379–14393, doi: 10.18653/v1/2024.emnlp-main.796.
[12] X. Ma, M. Strube, and W. Zhao, "Graph-based clustering for detecting semantic change across time and languages," in *Proc. EACL*, 2024, doi: 10.18653/v1/2024.eacl-long.93.
[13] A. Kutuzov, M. Fedorova, D. Schlechtweg, and N. Arefyev, "Enriching word usage graphs with cluster definitions," in *Proc. LREC-COLING*, 2024, pp. 6189–6198. [Online]. Available: https://aclanthology.org/2024.lrec-main.546/
[14] D. Loureiro, F. Barbieri, L. Neves, L. Espinosa Anke, and J. Camacho-Collados, "TimeLMs: Diachronic language models from Twitter," in *Proc. ACL System Demonstrations*, 2022, pp. 251–260, doi: 10.18653/v1/2022.acl-demo.25.
[15] F. Barbieri, J. Camacho-Collados, L. Neves, and L. Espinosa-Anke, "TweetEval: Unified benchmark and comparative evaluation for tweet classification," in *Findings of EMNLP*, 2020, pp. 1644–1650, doi: 10.18653/v1/2020.findings-emnlp.148.
[16] A. Longa et al., "Graph neural networks for temporal graphs: State of the art, open challenges, and opportunities," *Transactions on Machine Learning Research*, 2023. [Online]. Available: https://openreview.net/forum?id=pHCdMat0gI
[17] A. Kutuzov and M. Giulianelli, "UiO-UvA at SemEval-2020 Task 1: Contextualised embeddings for lexical semantic change detection," in *Proc. SemEval*, 2020, pp. 126–134, doi: 10.18653/v1/2020.semeval-1.14.
[18] E. Asgari, C. Ringlstetter, and H. Schütze, "EmbLexChange at SemEval-2020 Task 1: Unsupervised embedding-based detection of lexical semantic changes," in *Proc. SemEval*, 2020, pp. 201–207, doi: 10.18653/v1/2020.semeval-1.24.
[19] M. Martinc, S. Montariol, E. Zosa, and L. Pivovarova, "Discovery Team at SemEval-2020 Task 1: Context-sensitive embeddings not always better than static for semantic change detection," in *Proc. SemEval*, 2020, doi: 10.18653/v1/2020.semeval-1.6.
[20] A. Karnysheva and P. Schwarz, "TUE at SemEval-2020 Task 1: Detecting semantic change by clustering contextual word embeddings," in *Proc. SemEval*, 2020, pp. 232–238, doi: 10.18653/v1/2020.semeval-1.28.
[21] F. Periti, P. Cassotti, H. Dubossarsky, and N. Tahmasebi, "Analyzing semantic change through lexical replacements," in *Proc. ACL*, 2024, doi: 10.18653/v1/2024.acl-long.246.
[22] R. Kishino, H. Yamagiwa, R. Nagata, S. Yokoi, and H. Shimodaira, "Quantifying lexical semantic shift via unbalanced optimal transport," in *Proc. ACL*, 2025, doi: 10.18653/v1/2025.acl-long.774.
[23] R. Goworek and H. Dubossarsky, "Rethinking metrics for lexical semantic change detection," in *Proc. LChange*, 2026, pp. 147–161, doi: 10.18653/v1/2026.lchange-1.13.
[24] Z. Ling and M. Komachi, "LSC-Pattern: Interpretable lexical semantic change analysis via diachronic sense distributions," in *Proc. EMNLP*, 2026, pp. 6003–6020. [Online]. Available: https://aclanthology.org/2026.emnlp-main.264/