

Magdalena Nowakowska

Uniwersytet Łódzki

Potoczność jako źródło kreatywności językowej na blogach programistycznych

Ekspansja języka potocznego, o której pisze Małgorzata Kita [1991], dotyczy coraz szerszego zasobu tekstów. Formy zaczerpnięte z użycia, wykraczające poza normę wzorcową czy wytyczne gatunkowe, stają się komunikacyjnym standardem – zwłaszcza w przestrzeni internetu. Na potrzeby analizy języka blogów programistycznych za przejawy kategorii potoczności w języku uznaję wszystkie elementy leksykalne, składniowe oraz fleksyjne, które, pozostając poza wskazaną wyżej normą, noszą znamiona nieoficjalności, niestaranności oraz ekspresywnego nacechowania. Jest to zatem rozumienie bliskie zawartemu w pracach Andrzeja Markowskiego [1992] czy Jacka Warchali [2003], zbieżne z rejestrem swobodnym języka potocznego z jednej i typem emocjonalnym z drugiej strony w ujęciu Jerzego Bartmińskiego [2001]¹. Odróżniam przy tym zjawiska potoczne od kolokwialnych, wywodzących się z niskiego rejestru języka,

¹ Jerzy Bartmiński rozumie potoczność jako kategorię antropologiczną, a styl potoczny jako bazę dla wszystkich innych odmian języka narodowego, opartą na słownictwie wspólnoodmianowym. Wskazuje przy tym na podział języka potocznego na rejestry neutralny i emocjonalny oraz neutralny i emocjonalny [Bartmiński 2001]. Przyjęte w artykule rozumienie potoczności jest zatem zbieżne jedynie z częścią szerokiej definicji potoczności, wskazywanej przez badacza.

zwłaszcza z zakresie leksyki. Określenie **kolokwializm**, zestawiane przez Warchalę ze swobodną formą języka [Warchala 2003: 162], nazywa zjawiska silnie nacechowane emocjonalnie, związane z odmianą mówioną, które nie mają jednak tak negatywnych konotacji jak wulgaryzmy. W omawianym materiale kolokwializmy nie stanowią istotnej grupy.

Przeciwieństwem języka potocznego w zaprezentowanym ujęciu jest nie tylko język polszczyzny wzorcowej, ale także odmiana specjalistyczna, obsługująca obszerny zasób zapożyczeń z innych języków (zwłaszcza języka angielskiego), unikająca wartościowania oraz dążąca do precyzyjności komunikatów. Przykładami mogą stać się słownictwo i formy, jakimi posługują się informatycy oraz inne osoby związane z nowymi technologiami. Należy jednak wziąć pod uwagę, że część słownictwa specjalistycznego z tego zakresu uległa determinologizacji i została włączona do zasobu języka ogólnego [Maciołek 2013: 168]. Jest to spowodowane rozpowszechnieniem umiejętności i potrzeb korzystania z nowoczesnego sprzętu i oprogramowania. Dostęp do nowych technologii oraz obcowanie przeciętnego użytkownika komputera z terminologią specjalistyczną, połączone z intensywnym rozwojem blogosfery w Polsce sprawiły, że pojawił się popyt na blogowe poradniki informatyczne (w tym również programistyczne), pisane przez specjalistów. Jednak, co jest specyficzne dla tej formy publikacji treści, istnieje grono odbiorców, które regularnie czyta dany blog, nawet jeśli nie szuka w nim odpowiedzi na konkretne pytanie [Suska 2008: 140]. Uwaga ta dotyczy jednak tylko blogów zawężonych tematycznie, a nie internetowych, zmodyfikowanych dzienników [Gumkowska 2009: 231; Zajac i in. 2009: 219], nastawionych na chęć publicznego zwierzania [Suska 2008: 139]. Problem blogów specjalistycznych, w tym programistycznych, polega zatem na utrzymaniu czytelnika, który trafił na stronę przy okazji szukania porady technologicznej. Jest to podstawowe wyzwanie wszystkich treści popularnonaukowych.

Blogerom z pomocą przychodzą pomysły na równoważenie treści merytorycznych wpisami o sprawach prywatnych i okołozawodowych, związanych luźno lub niewiążących się wcale z głównym

tematem blogu (np. wycieczki, zarządzanie czasem, rozwój osobisty). Z uwagi na to, że to właśnie tekst w wymiarze językowym pełni w blogach najważniejszą funkcję [Gumkowska i in. 2009: 285], okazuje się, także w przypadku blogów eksperckich [Wierzbicka 2011: 177], że kluczem do zdobycia i utrzymania czytelnika jest odpowiedni język. Najpopularniejsze (najczęściej odwiedzane i komentowane) według wyszukiwarki Google w pierwszej połowie 2018 roku blogi programistyczne łączą treści merytoryczne z poczuciem humoru i kreatywnością językowych rozwiązań, które nie tylko pomagają średniozaawansowanemu internaucie zrozumieć poradę i rozwiązać problem, ale także mają zachęcić go do dłuższego pozostania na stronie, opatrzenia wpisu komentarzem [Hope 2013: 24] i powrotu za jakiś czas. Pojęcie kreatywności językowej rozumiem za Iwoną Kaproń-Charzyńską jako „zdolność użytkowników języka niezbędną do produkowania nowych i wartościowych wytworów, związaną z płynnością, giętkością i oryginalnością myślenia, albo przejawianie przez nich określonych form zachowania, polegających na zamierzonym użyciu narzędzi językowych niezgodnym z ich przeznaczeniem” [Kaproń-Charzyńska 2017: 179]. Obserwować zatem można nakładanie się na język blogów eksperckich dwóch zjawisk: precyzyjności wypowiedzi, narzucanej przez dyscyplinę, której dotyczą, a także emocjonalności i podobieństwa do języka mówionego [Szczerbał, Kamasa 2012: 123–124], specyficznych dla gatunku i komunikacji internetowej.

Do analizy włączonych zostało dziesięć popularnych (wskazywanych przez wyszukiwarkę Google jako często aktualizowane, odwiedzane i komentowane) blogów programistycznych, w których odnaleźć można przejawy kreatywności językowej. Są to (obok nazwy podaje stosowane w artykule akronimy): *Blog informatyczny* [BI] (dostępny pod adresem www.blog.kmagdziazr.pl); *Blog informatyczny* [BI2] (www.miroslawzelent.pl); *Blog o nauce programowania – Będę programistką* [BP] (www.bedeprogramistka.pl); *Cezary Walenciuk* [CW] (www.cezarywalenciuk.pl); *Ignormatyk* [I] (www.xpil.eu); *Itcraftsman – blog programisty rzemieślnika* [ICM] (www.itcraftsman.pl); *Kidaj blog – blogowanie na freelance* [KB] (www.kidaj.ad3.eu);

Na miękko [NM] (www.namiekko.pl); *Programistka* – Iwona Kubowicz [P] (www.programistka.com); *Z kodem przez świat* [ZK] (www.zkodemprzezswiat.pl)².

Najbardziej oczywistym przejawem kreatywności językowej są neologizmy. W analizowanym materiale uwagę zwracają okazjonalizmy i indywidualizmy, które są typowym zjawiskiem językowym dla blogów, niezależnie od kategorii tematycznej, do której należą. Swoje teksty poradnikowe szczególnie chętnie urozmaica nimi autor blogu *Ignormatyk*. Warto zauważyć zapowiadającą kreatywne podejście do języka nazwę tej strony, która została utworzona z połączenia leksemów *ignorować* oraz *informatyk*. Uwagę zwracają także inne działania z zakresu kreatywności językowej, takie jak derywacja paradygmatyczna. W jednym z postów tego blogera znaleźć można na przykład radę, by ułożyć elementy bazy, potrzebnej do wykonania polecenia, w kolejności motywowanej natężeniem jednej cechy, czyli „posortowane od najbardziejszego do najmniejbardziejszego” [I]. Autor tego okazjonalnego neologizmu paradygmatycznego, opartego na zmianie przysłówka w przymiotnik (*bardziej – bardziejszy*), uzyskał skrótość porady i efekt humorystyczny, przy jednoczesnym zachowaniu zrozumiałości wypowiedzi, kluczowej dla odbiorców blogu, szukających na stronie prostych rozwiązań nurtujących ich programistycznych problemów. Inny przykład analogicznego zjawiska to odsłonięcie procesu przekształcenia należącego do zasobu potocznego przysłówka *ciut* w równobrzmiący rzeczownik w cytacie: „Ciut w górę, dwa ciuty w lewo, może jednak trzy i pół ciuta w dół” [KB]. Celem zabiegu jest uzyskanie efektu humorystycznego i jednocześnie wyrażenie ironii względem zachowania, które niewiele wnosi do efektu końcowego pracy nad wyglądem strony, a zajmuje niepotrzebnie czas.

Innym przykładem okazjonalizmu tego samego autora jest „pytanie nizgruszzenizpietruszne” [I]. Przymiotnik, utworzony od potocznej frazy przysłówkowej *ni z gruszki, ni z pietruszki*,

² Dostęp do treści opublikowanych na podanych stronach internetowych: 15.05.2018.

oznaczającej tyle, co „niespodziewanie, zaskakując wszystkich” [WSJP], podobnie jak poprzednie przykłady niekonwencjonalnych rozwiązań leksykalnych, realizuje ludyczną funkcję neologizmów. Brak zmiany znaczenia omówionych okazjonalizmów w odniesieniu do form bazowych pozwala na właściwą interpretację derywatów. Przykładem okazjonalizmu jest również rzeczownik *sprawdzacz* [NM], użyty w bezpośrednim zwrocie do czytelnika: „bądź moim sprawdzaczem” – derywat sufiksalny, utworzony został od bazowego czasownika *sprawdzać* zgodnie z zasadami słowotwórczymi, analogicznie do innych nazw wykonawców czynności, takich jak *biegacz* czy *spawacz*. Jego utworzenie pozwoliło na zbudowanie krótkiego i jednocześnie nacechowanego humorystycznie wypowiedzenia. Analogiczny zabieg zastosowała inna blogerka w sformułowaniu swojego zawodowego celu: „być wymiataczem w JS i zaprzyjaźnić się z dziadem”. Neologizm *wymiatacz* pochodzi od potocznego i metaforycznego jednocześnie znaczenia czasownika *wymiatać* „być w czymś najlepszym”. Poza tym w przytoczonym fragmencie uwagę zwraca nazwanie języka programowania JavaScript „dziadem”, stanowiącym określenie na zjawisko nie lubiane, do którego podchodzi się z niechęcią. Nieliczne przykłady okazjonalizmów, mających swoje źródło w spontanicznym języku mówionym, pokazują potrzebę skrótowości i wprowadzenia do tekstu żartu językowego przy jednoczesnym zachowaniu jego jasności.

W badanym materiale znaleźć można także przykłady z pogranicza okazjonalizmów i zasobu idiolektalnego. Jeden z blogerów oznajmia: „instalacji GIMPA pod Windą nie muszą tłumaczyć” [BI], kiedy ma na myśli proces instalowania programu graficznego na komputerze wyposażonym w system Windows. Kreatywny zabieg językowy, wykorzystany przez autora, to kontekstowy neosemantyzm, którego frekwencji nie potwierdzają inne źródła informacyjne. Leksem *winda*, przez podobieństwo graficzne do angielskiej nazwy systemu operacyjnego, zostaje użyty jako jego nazwa, co bloger podkreśla, stosując zapis wielką literą. Inny przykład zjawiska leksykalnego, posiadającego cechy idiolektalne, to nazwanie

pendrive'a (*USB flash memory*) „flaszka”³. Uzasadnienie takiego kontekstowego neosemantyzmu tkwi, podobnie jak w powyższym przykładzie, w podobieństwie polskiego słowa „flaszka”, będącego potocznym określeniem butelki wysokoprocentowego alkoholu i angielskiego słowa *flash*, oznaczającego „błyskawiczny”.

Oprócz okazjonalnych neologizmów, materiał obfituje w określenia, w których przejawia się kreatywność leksykalna, należące jednocześnie do profesjolektu informatyków. Należy jednak zauważyć, że oprócz specjalistycznych określeń, autorzy poradników decydują się na wprowadzenie do swoich tekstów elementów zaczerpniętych z zasobu zawodowego języka mówionego. Niektóre z omówionych niżej sformułowań ulegają determinologizacji. W artykule zamieszczonym na jednej z poddanych analizie stron trafić można na powszechnie już rozumiane określenie „pliki puchną” [KB]. Przyrastająca w sposób niepożądany przez ich twórcę waga dokumentów elektronicznych zostaje nazwana leksemem używanym w dosłownym znaczeniu w odniesieniu do niewłaściwej pracy ludzkich tkanek [WSJP], ale rozszerzenie semantyczne jest oparte na analogii i przez to zrozumiałe. Podobnie inne określenie, także czerpiące ze słownictwa nacechowanego potocznie, „zarzynanie CPU” [KB], znaczące tyle, co „nadmierne obciążanie procesora, mogące go uszkodzić”, jest zbudowane analogicznie do innych, obecnych w odmianie potocznej, w których występuje leksem *zarzynać* (np. w odniesieniu do silnika samochodu). Zawarty w tym znaczeniu element gwałtownego zachowania, związanego z przemocą, pochodzi od pierwotnego pola semantycznego – zabijania z użyciem ostrego narzędzia [WSJP]. Przechodzące do języka ogólnego są określenia *odpalić plik* [NM] (na innym blogu: „teraz możemy już odpalać!” [ZK]), oznaczający tyle co go „uruchomić, otworzyć (z potocznym nacechowaniem)” czy *wyklikać* („wyklikać swoje wymagania w przeglądarce” [NM]), będący derywatem prefiksальnym pochodzącym od leksemu *klikać*, czyli „przyciskać

³ „Wziąłem flaszkę. Nie, nie zacząłem pić alkoholu z rozpacz. Flaszkę, czyli pędzaka. Takiego na USB. Pendrive, w sensie” [I].

klawisz komputerowej myszy w celu wydania określonej komendy” [WSJP]. Znaczenie leksemu *wyklikać* jest jednak szersze niż tylko zmiana aspektu względem wyrazu podstawowego. Oznacza „celowe i schematyczne wydawanie komend z użyciem myszy” i, podobnie jak *odpalać*, jest leksemem z zasobu języka potocznego. Do języka ogólnego, ze względu na zwiększającą się dostępność specjalistycznego sprzętu i zaawansowanego oprogramowania, przechodzi także określenie „wrzucać na serwer”, związane z częścią publikacją, umożliwiającą testowanie i następnie wdrożenie na przykład tworzonej aplikacji: „wrzucić swój kod na serwer” [P]. Kreatywność leksykalna specjalistycznego słownika programistów przejawia się także w innych określeniach. Jednym z nich jest nazwanie zawilego kodu „kodem spaghetti” [ICM] – metafora, stosowana w potocznym języku programistów, opiera się na podobieństwie desygnatów: niepoprawnie skonstruowany kod jest trudny do prześledzenia, podobnie jak poplątane są nitki ugotowanego makaronu spaghetti.

W języku programistów można zauważyć, podobnie jak we wszystkich innych odmianach języka, tendencję do ekonomizacji, przejawiającą się w tworzeniu sformułowań skrótowych. Typowe dla mowy potocznej są uniwerbizmy. W języku programistów-bloggerów przykładami z tego zakresu leksyki są na przykład: *makfani* [NM] czyli „fani komputerów Mac, produkowanych przez koncern Apple” czy *dev* [BP], zastępujące określenie „software developer”, czyli „twórca oprogramowania”. Występujące bez dodatkowych elementów wyjaśniających są także rzeczowniki odliczebnikowe, działające na zasadzie uniwerbizmów, takie jak *siódemka* [BI2] czy *ósemka* („po Siódemce przyszła Ósemka” [I]), oznaczające kolejne wersje oprogramowania, najczęściej systemu Windows, chyba że kontekst wskazuje inaczej.

Blogerzy-programiści korzystają z bogatego zasobu leksyki potocznej, żeby urozmaicić artykuły instruktażowe o elementy nie tylko wywołujące efekt humorystyczny przez stosowanie kontrastu, ale także zbliżające hermetyczny profesjolekt do języka używanego powszechnie. Jest to powód, dla którego zamiast *wdrożenia*

czy publikacji można przeczytać, że „CMS wypuszcza się w świat” [KB]. Ten sam autor wyjaśnia, że „ma świra na punkcie optymalizacji” [KB], używając sytuującego się stojącego na pograniczu języka potocznego i kolokwialnego zwrotu: *mieć świra na jakimś punkcie* w odniesieniu do działania nazwanego zgodnie z profesjolektem, mającego na celu udoskonalenie kodu (*optymalizacja*). Podobne zastosowanie potocznego zwrotu w języku tego blogera znaleźć można w wypowiedzi: „próby przemycenia niebezpiecznego kodu idą na spacer” [KB], w którym ponownie zestawione zostały sformułowania zaczerpnięte z programistycznego profesjolektu (*niebezpieczny kod* to taki, który jest obciążony ryzykiem wystąpienia błędu) ze sformułowaniami typowymi dla języka potocznego: *coś poszło sobie (na spacer)* znaczy tyle, co „zostało zakończone niepowodzeniem”.

Lektura wpisów na badanych blogach przynosi obszerny materiał z zakresu kreatywnego wykorzystania potocznych. Za przykład posłuży tu wyrażenie „cud, miód i orzeszki”, interpretowane jako zespół pozytywnych cech, odbieranych różnymi zmysłami (nie tylko smakiem), zestawiony przez autorkę jednego z analizowanych blogów z udogodnieniami oferowanymi przez język oprogramowania: „Flexbox to jest taki cud, miód i orzeszki” [NM].

Blogerka zdecydowała się na skorzystanie z zasobu potoczności, żeby skonstruować tekst wyróżniający się pod względem płynności budowania porównań, nadających mu wrażenia większej przystępności dla niezaawansowanego odbiorcy, a tym samym popularnonaukowy. Równie istotnym czytelnikiem, dla którego powstają takie zaskakujące połączenia potocznych i profesjonalizmów, jest osoba, która przypadkowo trafiła na stronę i być może na nią wróci, jeśli zainteresuje ją sposób przekazywania porad. Kluczem może się okazać właśnie nieszablonowość, taka jak ta widoczna w wyżej przytoczonym fragmencie.

Zdarza się jednak, że programista, który na co dzień korzysta z profesjolektu, staje przed zadaniem wyjaśnienia przypadkowemu odbiorcy lub początkującemu programiście pewnych pojęć i to zadanie go przerasta. Przykładem może być wypowiedź z blogu *Ignor-matyk*, będąca zapisem takich trudności: „Systemy źródłowe to nic

innego jak, khem, jak by to ująć... no, systemy źródłowe” [I]. Czytelnik widzi nie tylko nieudaną próbę zdefiniowania zjawiska, ale także zapis świadczący o trudności, z jaką zmierzył się bloger. Zastanawianie się i wątpliwości, obecne często w języku mówionym, są pomijane w zapisie. W podanym przykładzie zostały wyeksponowane cechy, które nie tylko nie współgrają z elementami profesjonalnymi, ale także z formą pisaną, jaką jest artykuł na blogu.

Podobnie, choć mniej wyraźnie, przejaw tej tendencji zaobserwować można w treści wpisów na innym blogu: „hmm jakby to określić...” [BP]. Aspekt humorystyczny współgra z chęcią programisty, by stworzyć tekst nieszablonowy, wyglądający tak, jakby był napisany spontanicznie i momentalnie publikowany.

Wydaje się, że to właśnie ten ostatni czynnik – wrażenie spontaniczności – sprawia, że elementy językowe z zasobu potoczności nie należą do rzadkich w artykułach opublikowanych w blogach programistycznych. Obok wyrażen nacechowanych potocznie, ale posiadających swoje źródło w profesjolekcie (jak choćby omówiony wyżej neosemantyzm *odpalać*), programiści korzystają z określeń, których głównym zadaniem jest zwiększenie przystępności tekstu i wywołanie efektu humorystycznego, a przy okazji wyróżnienie się na tle innych autorów, którzy blogują w obrębie tej samej kategorii. Zabawnie napisany tekst zwiększa prawdopodobieństwo dłuższego przebywania na stronie przez internautę, a to z kolei wpływa na poprawę statystyk wyświetleń i ewentualny wzrost zarobków blogera. W artykułach zamieszczonych na badanych blogach znaleźć można formy zaczerpnięte z języka potocznego, takie jak wyrażający odczucia autora zwrot „w sumie nie wiem” [KB] czy zawierający ocenę „trochę dziwne” [KB]. Rozbudowany jest zasób leksemów użytych w kontekstach wskazujących na wykorzystanie potocznego znaczenia, np. „ogarnąć Twittera” [KB], „[fanpage na Facebooku] zdechł śmiercią naturalną” [KB], „totalnie rozwała mnie, kiedy poważny dokument jest napisany Comic Sansem” [KB], „jeśli nie chcesz się bawić w komendy” [CW], „przerzucam się na pracę w środowisku Eclipse” [NM], czy „kombinacja klawiszy, o której wiadomo, że wysadzi wszystko” [NM]. Za sprawą takich zwrotów (*ogarniać*,

zdechnąć, rozwalać, bawić się w, przerzucić się, wysadzić) język blogów sprawia wrażenie bardziej przystępnego, a sam bloger jawi się jako przeciętny użytkownik polszczyzny, bliski pod tym względem czytelnikowi blogu.

Potoczność w języku blogów obejmuje także inne niż czasowniki części mowy, np. *fajny* [BI2], *niekumaty* [CW] – „taki, który niewiele rozumie”, *pi x oko* [I] – „mniej więcej”, *świeżak* [ICM] – „osoba początkująca”, *nakręcanie szalu* [CW] – „wywoływanie zainteresowania”, *jak gdyby nigdy nic* [BI2]. Takie podejście, łączące elementy języka mówionego z językiem specjalistycznym, pozwala zbudować tekst w pewnym zakresie kreatywny, wykraczający poza schematy odmian językowych – zarówno potocznej, jak i specjalistycznej.

Artykuły zamieszczone przez programistów na blogach są przede wszystkim specjalistycznymi poradnikami dla osób średnio zaawansowanych, które chcą samodzielnie rozwiązać problem techniczny. Dlatego ważne jest, także z uwagi na wywoływanie reakcji w postaci udostępnień i komentarzy, podwyższających pozycję w wyszukiwarce Google, utrzymywanie kontaktu z czytelnikiem. Funkcja fatyczna, typowa dla języka mówionego, przejawia się w określeniach, odwołujących się właśnie do procesu mówienia, a nie pisania, ponadto także do kontaktu osobistego, niezapośredniczonego. Przykładami są zwroty: „brzmi nieźle, prawda?” [ICM], „zapinajcie pasy i siadajcie, zaczynamy” [ICM], „ej, zaraz, co jeśli Java idzie złą ścieżką?” [CW], „w tym shorcie pojawia się JavaScript – pogadamy o różnicy pomiędzy definiowaniem zmiennych klauzulą *var* oraz z użyciem nowszej klauzuli *let*” [BI2] czy „rozumiesz zasadę? Łatwizna, prawda?” [BP]. Ich funkcją jest także porządkowanie wyводу, utwierdzenie się w przekonaniu, że porada jest zrozumiała oraz realizacja potrzeby dialogiczności, występującej w języku mówionym. Nieco szerszą funkcję, bo uzupełnioną o ironię, pełni przytoczone, potencjalnie wypowiedziane przez innego programistę pytanie o zmniejszanie wagi utworzonego kodu przez łączenie i usuwanie jego warstw: „a może by tak urwał, od razu siedemnaście, albo pińcet, hy?” [I]. W zapisie liczebnika ze spłaszczeniem samogłoski nosowej (*pińcet*) oraz nieco niegrzecznym pytaj-

niku (*hy?*) mieści się opinia blogera o braku przekonania co do sensu nadmiernego skracania kodu przez zaburzanie odrębności funkcji przez niego realizowanych. Warto zwrócić uwagę na leksem *urwać* („dostać, załatwić, usunąć”), sytuujący się na pograniczu niestaranego rejestru języka potocznego i języka kolokwialnego.

Blogerzy-programiści, przybliżając użytkownikowi internetu swoją pracę, wykorzystują porównania do zjawisk, które odbiorca dobrze zna. Ma to na celu, obok obrazowości i zwiększenia zrozumiałości treści, także wytworzenie przekonania, że czytelnik poradzi sobie z zadaniami stawianymi przed nim przez autora blogu. Kiedy jeden z blogerów wyjaśnia, co jest kluczowe, by kod był poprawny, zapisuje: „Nie możemy stworzyć fabryki, która produkuje samochody i jednocześnie szyje buty” [ICM]. Porównanie typu kodu, tak zwanej klasy spełniającej jedno proste zadanie, do fabryki, ma na celu uświadomienie wagi decyzji programisty, do czego dany fragment kodu jest potrzebny. Inny przykład daje autor blogu *Z kodem przez świat*: „Jest to kombajn do zarządzania użytkownikami. No właśnie – kombajn. Wielka kobyła, która sama w sobie stanowi dość sporą aplikację” [ZK]. Znalezienie analogii pomiędzy gotową aplikacją zarządzającą kontami użytkowników a kombajnem daje wyobrażenie o zużyciu miejsca i energii, a także wątpliwej wygodzie korzystania z niej. Potoczna jest także, oprócz przywołanego wycinka rzeczywistości, forma krytyki takiego gotowego programu przez nazwanie go *kobyłą*. Przykładem zestawienia rzeczywistości potocznej i technicznej jest także opis funkcjonalności, na jaki zdecydowała się inna autorka blogu: „możemy kontrolować, jak mają zachowywać się dzieci naszego kontenera” [BP]. *Kontenerem* został tu nazwany system klas, natomiast same klasy, wykonujące proste obliczenia i zadające pytania z dychotomicznymi odpowiedziami, zostały określone jako jego *dzieci*.

Blogujący programiści rzadko natomiast używają wulgaryzmów. Jeden z nielicznych przykładów to derywat rzeczownikowy od wulgarnego czasownika użyty w sformułowaniu: „bez ryzyka totalnej rozpiarduchy” [I]. Częściej jednak pojawiają się eufemizmy wyrazów wulgarnych, np. zamiast leksemu *dupa* autorzy (w tych

przypadkach autorki) decydują się na użycie synonimicznych wyrazów – *pupa* i *tyłek*: „zaczęło nam się palić pod pupami” [NM] czy „spinam tyłek i uczę się intensywnie” [BP]. Wulgaryzm *dupa* występuje z dopowiedzeniem, które ma na celu usprawiedliwienie i zaznaczenie, że autorka ma świadomość nieprzyzwoitości wyrazu: „kolokwialnie mówiąc – dałam trochę dupy” [BP].

Omawiając zjawisko kreatywnej potoczności w artykułach zamieszczonych na blogach programistycznych, nie można pominąć dłuższych fragmentów, mających na celu zaskoczenie odbiorcy żartem językowym, takich jak np.: „obejdzie się bez całek, trygonometrii i bardziej zaawansowanych zwierząt” [I], w którym w jednej grupie zestawione zostają funkcje i obliczenia matematyczne oraz żywe istoty. Składnia tego wypowiedzenia sugeruje, że całki i trygonometria także są elementami fauny. Określenie „zaawansowane zwierzęta” na skomplikowane zjawiska ze świata nieożywionego znajduje potwierdzenie w uzusie. Innym, jeszcze ciekawszym przykładem, jest wypowiedź tego samego blogera: „Teraz pracowicie ściubię kolejny tysiąc-z-hakiem. Jeżeli, Czytelniku, podczas lektury mojego blogu słyszysz jakieś stukanie, to pewnie ten hak” [I]. Na uwagę zasługuje zarówno zapis z łącznikami określenia „tysiąc-z-hakiem”, znaczącego tyle, co 1024 (liczba jest konsekwencją stosowania w informatyce systemu dwójkowego zamiast dziesiętnego), jak również defrazeologizacja przez użycie leksemu *hak* w znaczeniu przenośnym („trochę więcej niż”) i dosłownym (metalowego, wiszącego przedmiotu, który może wywoływać dźwięk stukania).

Inny humorystyczny przykład zaczerpnięty z blogu *Ignormatyk* jest oparty na zestawieniu sfery skojarzeń erotycznych z opisem dobrze skonstruowanego kodu: „Niestety, naga prawda jest taka, że po pierwsze primo będę dziś promował krótkość, a nie długość, a po drugie primo nie wiem, o czym tam sobie jeden z drugim myśli, w każdym razie chodzi dziś o długość (a raczej: krótkość) kodu realizującego pewne całkiem niebanalne zadanie” [I]. Uruchomienie skojarzeń z pogranicza wulgarności następuje nie tylko przez ustawienie w bliskiej kolokacji leksemów *długość* oraz *nagi* z wyrażenia *naga prawda*. Nie bez znaczenia jest również zaczerpnięty z mowy

potocznej komentarz „nie wiem, o czym tam sobie jeden z drugim myśli”, potwierdzający intencję autora tych słów, by zostały one niejednoznacznie odebrane. Warto zwrócić uwagę na przynależne do niestarannej mowy potocznej sformułowania z zaburzoną logiką: „po pierwsze primo” i „po drugie primo”, stanowiące kontaminację zwrotów „po pierwsze... po drugie...” oraz „*primo... secundo...*” i będące jednocześnie żartobliwym odniesieniem do języka potocznego.

O ile wykorzystanie w analizowanych treściach anglicyzmów jest uzasadnione potrzebą nazwania zjawisk z zakresu programowania, o tyle na uwagę zasługuje ich kreatywne wykorzystanie, którego przykładem może być wypowiedź autorki blogu *Będę programistką* na temat nauki jednego z zagadnień: „póki co Backend to dla mnie zupełna czarna magia. Nie czuję z nim *flow*. Nie czuję też chemii między mną a Backendem” [BP]. Składająca się z trzech krótkich wypowiedzi konstatacja stanowi zaskakujące zestawienie leksykalne, będące połączeniem profesjonalnej nazwy (*Backend*), polszczyzny potocznej (wyrażenie frazeologiczne *czarna magia*), kontaminacji frazeologizmów *czuć miętę do* i *jest chemia między*, a także zwrotu angielskiego (*flow*), wykorzystanego do zbudowania zdania powtarzającego treść wypowiedzi stojących obok.

Język blogów programistycznych nie wyróżnia się spośród innych stron pod względem obecności potocznych. „Jednak w odniesieniu do wyspecjalizowanych porad, zdarza się, że obecność słownictwa nacechowanego jest rażąca. Jego wysoka frekwencja pozwala stworzyć listę przyczyn, dla których autorzy blogów informatycznych sięgają po leksykę z zasobu potoczności. Dążenie do efektu humorystycznego i zrozumiałości treści jest niezbędne, by utrzymać satysfakcjonującą liczbę wyświetleń blogu. Ponadto język analizowanych stron charakteryzuje się nieszablonowymi zjawiskami leksykalnymi, które świadczą o wyróżniającej się świadomości językowej ich autorów i przekonaniu o wpływie sposobu wypowiedzenia się na sukces blogerski. Nie bez znaczenia jest także zamysł, choć niewyraźny *explicite*, by artykuły miały walor popularnonaukowy, którym rządzi – między innymi – przystępność języka wypowiedzi.

Wykaz skrótów

- Blog informatyczny* [BI], www.blog.kmagdziarz.pl [dostęp: 15.05.2018].
- Blog informatyczny* [BI2], www.miroslawzelent.pl [dostęp: 15.05.2018].
- Blog o nauce programowania – Będę programistką* [BP], www.bedeprogramistka.pl [dostęp: 15.05.2018].
- Cezary Walenciuk* [CW], www.cezarywalenciuk.pl [dostęp: 15.05.2018].
- Ignormatyk* [I], www.xpil.eu/ [dostęp: 15.05.2018].
- Itcraftsman – blog programisty rzemieślnika* [ICM], www.itcraftsman.pl [dostęp: 15.05.2018].
- Kidaj blog – blogowanie na freelance* [KB], www.kidaj.ad3.eu [dostęp: 15.05.2018].
- Na miękko* [NM], www.namiekko.pl [dostęp: 15.05.2018].
- Programistka – Iwona Kubowicz* [P], www.programistka.com [dostęp: 15.05.2018].
- Z kodem przez świat* [ZK], www.zkodemprzezswiat.pl, [dostęp: 15.05.2018].

Bibliografia

- Bartmiński J. [2001], *Styl potoczny*, [w:] *Współczesny język polski*, Bartmiński J. (red.), Wydawnictwo Uniwersytetu Marii Curie-Skłodowskiej, Lublin.
- Gumkowska A. [2009], *Blogi wobec tradycji diarystycznej. Nowe gatunki w nowych mediach*, [w:] *Tekst (w) sieci*, t. 1, Ulicka D. (red.), Wydawnictwa Akademickie i Profesjonalne, Warszawa.
- Gumkowska A., Maryl M., Toczyski P. [2009], *Blog to... blog. Blogi oczyma blogerów. Raport z badania jakościowego zrealizowanego przez Instytut Badań Literackich PAN i Gazeta.pl*, [w:] *Tekst (w) sieci*, t. 1, Ulicka D. (red.), Wydawnictwa Akademickie i Profesjonalne, Warszawa.
- Hope E. [2013], *Blogi jako platforma komunikowania na przykładzie blogów uczelnianych*, „*Studia Ekonomiczne*”, nr 157, s. 22–33.
- Kaproń-Charzyńska I. [2017], *Kreatywność językowa i jej wybrane mechanizmy w kontekście wtórnej oralności*, „*Prace Filologiczne*”, nr 71, s. 177–186.

- Kita M. [1991], *Ekspansja potoczności*, „Prace Językoznawcze”, nr 19, s. 83–90.
- Maciołek M. [2013], *Adaptacja terminów informatycznych w języku ogólnym i artystycznym*, „Biblioteka Postscriptum Polonistycznego”, nr 3, s. 167–182.
- Markowski A. [1992], *Leksyka wspólna różnym odmianom polszczyzny*, Wiedza o Kulturze, Wrocław.
- Suska D. [2008], *O wybranych strategiach komunikacyjnych w blogu – (auto)reklamowość*, [w:] *Współczesna polszczyzna. Stan, perspektywy, zagrożenia*, Cygal-Krupa Z. (red.), Księgarnia Akademicka, Kraków–Tarnów.
- Szczerbal M., Kamasa V. [2012], *Język blogów a normy poprawnościowe języka polskiego*, „Investigationes Linguisticae”, nr 26, s. 123–153.
- Warchala J. [2003] *Kategoria potoczności w języku*, Wydawnictwo Uniwersytetu Śląskiego, Katowice.
- Wierzbicka A. [2011], *Od blogu do mikroblogu – (r)ewolucja gatunku?*, [w:] *Polszczyzna trzech pokoleń*, Wojtczuk K., Jasińska M. (red.), Wydawnictwo Uniwersytetu Przyrodniczo-Humanistycznego, Siedlce.
- Współczesny słownik języka polskiego* [2007], Dunaj B. (red.), Langenscheidt, Warszawa.
- Zając J., Rakocy K., Nowak A. [2009], *Interaktywne, choć osobiste blogi i blogowanie a komunikacja z otoczeniem*, [w:] *Tekst (w) sieci*, t. 1, Ulicka D. (red.), Wydawnictwa Akademickie i Profesjonalne, Warszawa.