Further developing a word mark similarity measurement framework – Part II: Defining an improved similarity score
by David Barnett

Introduction

My initial study on mark similarity measurement[1] focused on formulations for quantifying the objective similarity of pairs of marks, with particular focuses on colour- and word marks. As discussed in previous articles in this series, mark similarity assessment is a key part of the resolution of many intellectual property disputes, and a more objective approach could have a number of advantages, including the potential to provide definitions which could be built into case law, offer greater consistency across dispute decisions, and specify thresholds for IP protection.

However, it is important to reiterate the key point that any objective algorithms of these types should only ever be considered as *tools* to be used as part of the overall assessment process, which overall includes significant degrees of *subjectivity*. In the first instance, the algorithmic frameworks presented in this series for word marks focus only on visual (spelling) and aural (pronunciation – with a specific basis in American English) similarity, with no account taken of conceptual similarity (i.e. meaning) or the influence of any associated logos, imagery or mark stylisation. Overall, dispute decisions are often reliant on an assessment of the *likelihood of confusion* between the marks in question, which is generally also dependent on a range of other factors, including the distinctiveness, degree of overlap of associated goods and services, strength and degree of renown of the marks, documented evidence of actual confusion, and the degree of attention paid by a typical consumer – many of which may vary between different geographical regions[2,3]. Some of the factors generally considered for the components which can be measured algorithmically (such as typically putting greater weight on comparisons between elements appearing at the start of the marks in question, and greater emphasis on differences appearing within shorter marks[4]) can, and have, been built into the proposed algorithms wherever possible.

The degree of similarity (of each *type*) between marks is often specified in dispute cases as 'high', 'medium' or 'low'; with this in mind, it seems reasonable (where constructing any measurement algorithm) to formulate the output as a similarity *score* (as proposed for colour marks in the previous article[5] in this series), which aligns broadly with this framework but offers a more quantitative basis for comparison (though keeping in mind that all of the above caveats also still apply!).

Formulation of the similarity score algorithm

The similarity score used for comparison of pairs of word marks ($\boldsymbol{S_{wor}}$), in both the previous study and this follow up, reflects both visual (spelling) and aural (pronunciation) similarity (only).

As in the initial version, *visual* similarity between the marks (i.e. in terms of their spelling) is quantified using two distinct algorithms, each of which reflects different aspects of the similarity. The two algorithms (each of which generates a score which can be expressed as a percentage) are:

---

[1] https://circleid.com/posts/towards-a-quantitative-approach-for-objectively-measuring-the-similarity-of-marks
[2] https://bowmanslaw.com/insights/degrees-of-similarity-put-to-the-test/
[3] https://www.taylorwessing.com/en/insights-and-events/insights/2021/03/were-confused-how-the-general-court-decides-when-trade-marks-are-confusingly-similar
[4] https://guidelines.euipo.europa.eu/1803468/1787906/trade-mark-guidelines/3-5-conclusion-on-similarity
[5] 'Further developing a colour mark similarity measurement framework – Part II: Defining a similarity score'

- The *fuzz.ratio* metric ($\boldsymbol{F_{lev}}$), an algorithm implemented in the Python package '*fuzzywuzzy*'[6], based on the concept of Levenshtein distance – a way of quantifying the number of edits required to transform one string into the other – but also taking account of other factors (including the length of the strings).

- The Jaro-Winkler similarity algorithm (and score ($\boldsymbol{sim_j}$)) (as implemented in the the Python package '*Levenshtein*'[7]), which includes an element of consideration of the proximity of the matching / non-matching characters to the *start* of the strings.

In the simplest formulation of the overall algorithm (and as retained here), the score component reflecting overall visual similarity ($\boldsymbol{S_{vis}}$) is expressed just as the simple mean of the above two scores (as below), although it would be possible to apply different weightings if required.

$$\boldsymbol{S_{vis}} = (\boldsymbol{F_{lev}} + \boldsymbol{sim_j}) / 2$$

For *aural* similarity, the proposed calculation framework is based on the creation of a phonetic representation of the marks / strings in question, and then a comparison of these representations (again, using the *fuzz.ratio* metric).

The initial formulation also made use of two distinct algorithms for generating the phonetic representations, based on the Soundex and NYSIIS (New York State Identification and Intelligence System) encodings. However, both of these have certain shortcomings, not least the poor handling of vowel sounds within the strings, and (in Soundex) the inability to encode any consonants beyond the first four.

In this improved version, therefore, I instead propose the use of the *Phonemizer* algorithm[8,9] for generating the phonetic versions of the strings, which utilises IPA (International Phonetic Alphabet)[10] encoding, and which was explored in the previous follow-up study[11] and appears to perform well (although some data 'cleansing' is required in some cases, to ensure that the algorithm interprets the string as intended). The aural similarity score ($\boldsymbol{S_{aur}}$) can then be calculated simply as the output of the *fuzz.ratio* metric applied to the IPA representations as given by *Phonemizer*, i.e.:

$$\boldsymbol{S_{aur}} = \boldsymbol{F_{Pho}}$$

As in the previous formulation, the overall (word mark) similarity score can then most simply be expressed just as the mean of the two individual components, i.e.:

$$\boldsymbol{S_{wor}} = (\boldsymbol{S_{vis}} + \boldsymbol{S_{aur}}) / 2$$

Similarity scores for test-pairs of marks

As an illustration of the performance of this algorithm, I consider a set of approximately 200 pairs of word marks, mostly the subjects of recent trademark disputes (several of which were also considered in previous articles in this series), and with a primary focus on single-word marks (for

---

[6] https://pypi.org/project/fuzzywuzzy/

[7] https://rapidfuzz.github.io/Levenshtein/levenshtein.html#jaro-winkler

[8] M. Bernard and H. Titeux (2021). 'Phonemizer: Text to Phones Transcription for Multiple Languages in Python', *J. Open Source Software*, 6(68), p.3958.

[9] https://pypi.org/project/phonemizer/

[10] https://www.internationalphoneticassociation.org/content/ipa-chart

[11] https://circleid.com/posts/further-developing-a-word-mark-similarity-measurement-framework

simplicity). The full set of mark-pairs, and the calculated similarity scores, are presented in Appendix A.

The first point to note is that, generally, little pre-processing of the data is required in order to utilise the algorithm. All marks have been converted to lower-case, though this is generally a matter of choice, just to ensure that upper- and lower-case versions of the same letter are treated identically. The algorithms do also appear to correctly handle accented characters (albeit that the phonetic representations will generally reflect an *English* pronunciation). The only two modifications to the data required in these cases were a rewriting of 'OrangeryOS' as 'orangery-o-s' (to ensure that the pronunciation is rendered as '*oh-es*') and (as in a previous study) of 'likeme' to 'like-me'.

Elsewhere (as noted previously), the *Phonemizer* algorithm renders 'unreadable' strings as individual characters (e.g. 'immun44' as '*immun-four-four*', '007' as '*zero-zero-seven*', 'ch_t.' as '*see-aitch-tee*', and 'mbfw' as '*em-bee-ef-doubleyu*'), though these versions have been retained in an unmodified state in the analysis. Some of these representations *may* not be as originally intended when the marks were conceived, however – e.g. 'genv3rse' is rendered as '*genv-three-rse*' (rather than the more likely '*genverse*'), and 'm4tter' as '*em-four-tter*' (rather than '*matter*').

Overall, however, the algorithm does seem to provide a (subjectively) reasonable ranking of the mark-pairs by similarity. An attractive additional characteristic of this framework is that it is entirely repeatable, and unreliant on the number and types of pairs in the dataset (i.e. a particular word-pair will *always* give the same score), so it is always possible to compare like-with-like. Accordingly, it is instructive to consider some representative examples of word-pairs giving particular (approximate) scores ($\boldsymbol{S}_{wor}$), to provide a 'reckoner' of what the scores represent, i.e.:

- Approx. 90%:
  - boss / bossi
  - billionaire / zillionaire
  - thermacare / thermocare
  - prinker / prink
  - intellicare / intelecare
  - chooey / chooee
  - mahendra / mahindra

- Approx. 80%:
  - zara / zarzar
  - rabe / rase
  - retaron / retlron
  - createme / create.
  - spa / spato
  - thermomix / termomatrix

- Approx. 70%:
  - kelio / kleeo
  - terry / terrissa
  - tygrys / tigris
  - nike / nuke

- Approx. 60%:
  - nutella / mixitella
  - airbnb / francebnb

- gallo / rampingallo
- iphone / mifon
- joy / bjoie
- jd / jdyaoying

- Approx. 50%:
  - zara / zorazone
  - quirón / quiromasté

- Approx. 40%:
  - book / restaubook
  - h10 / motel 10

An additional attractive aspect of this approach is that it is also possible, if required, to consider the visual and aural similarity components *separately*. For example, the top pairs of marks by *visual* similarity score ($\boldsymbol{S_{vis}}$) (<u>only</u>) are fashiongo / fashionego (96.50%), configon / configo (95.25%) and casoria / castoria (95.04%), and by aural similarity score ($\boldsymbol{S_{aur}}$) (only) are sanytol / sanitol, testex / test-x, hobbit / hobbyt , kramer / cramer, kresco / cresco, and cylance / sylence (all 100%, i.e. deemed phonetically identical).

Discussion

Overall, (and again as noted previously) it would not be reasonable to expect any significant correlation between the similarity scores and the *findings* reached in the associated disputes, because of the significant additional (and subjective) points considered in the analysis, as discussed in the introduction to this article. For example, in the Initio / Vinicio case, the marks were found to have 'below average' visual similarity (despite the quantitative objective visual similarity score of 80.96%), with consideration having been given in the case to the differing impact of the various elements and the overall impression of the respective marks, which feature significant differences in visual presentation[12].

Nevertheless, the similarity score does offer a useful tool to consider the 'pure' visual and aural similarity (only) of the word marks, as part of an overall analysis (for example, in dispute cases), in a framework which is repeatable and qualitative, providing the potential for a consistent approach to assessment of these characteristics. It also aligns with the familiar terminological descriptions of 'degrees' of similarity, whilst offering a more granular and continuous scale.

The algorithm does also offer additional possible use-cases, such as (for example) the ability to post-process the outputs from trademark watching services, so as to better sort the results by relevance (in cases where the sorting algorithm offered by the service performs less satisfactorily), and thereby aid in the review process.

It is also worth noting that there is also scope for possible future enhancements to the algorithms (some of which have been discussed previously), including (for example) assessments of the distinctiveness of the various elements or sub-elements (subsequences or substrings) of the marks, re-weighting the contribution of any trailing 's', and so on. Distinctiveness and analysis of the 'types' of elements present in the marks may, in particular, be key to making a more meaningful overall assessment of similarity and, ultimately, likelihood of confusion. Relevant examples for consideration in the dataset include Cylance / Sylence (both 'clearly' allusions to the same common

[12] Stobbs CaseFest #16, London, 02-Oct-2024

word (‘silence’)), Doctolib / Avocatlib (where the first portion of each mark makes reference to a profession), BMW / BMV (where the only difference is manifested as a pair of ‘similar’ letters), Immun44 / Immuno-19 (both featuring a similar root and, unusually, followed specifically by a number), iPhone / Mifon (with the similarity between ‘I’ and ‘me’ being of potential relevance), and Align / Clickalign (relevant because of the range of additional names cited by the latter party, suggesting the key point is the question of the distinctiveness of the term ‘align’ for the relevant goods and services).

Appendix A: Pairs of marks and their visual, aural and overall similarity scores

Mark 1	Mark 2	Vis. sim. score (S_{vis})	Mark 1 (IPA)	Mark 2 (IPA)	Aur. sim. score (S_{aur})	Overall word mark sim. score (S_{wor})
casoria	castoria	95.04	kæsoːɹiə	kæstoːɹiə	95.00	95.02
sanytol	sanitol	89.67	sænɪtɑːl	sænɪtɑːl	100.00	94.83
testex	test-x	88.17	tɛstɛks	tɛstɛks	100.00	94.08
hobbit	hobbyt	88.17	hɑːbɪt	hɑːbɪt	100.00	94.08
replay	re:play	94.10	ɹiːpleɪ	ɹiː pleɪ	94.00	94.05
kramer	cramer	85.94	kɹeɪmɚ	kɹeɪmɚ	100.00	92.97
kresco	cresco	85.94	kɹɛskoʊ	kɹɛskoʊ	100.00	92.97
cintra	citra	93.28	sɪntɹə	sɪtɹə	92.00	92.64
dekton	deton	93.28	dɛktən	dɛtən	92.00	92.64
free	freen	92.50	fɹiː	fɹiːn	91.00	91.75
goddess	godless	89.67	gɑːdəs	gɑːdləs	93.00	91.33
boss	bossi	92.50	bɔs	bɔsi	89.00	90.75
billionaire	zillionaire	92.47	bɪliənɛɹ	zɪliənɛɹ	89.00	90.73
thermacare	thermocare	91.89	θɜːmɐkɛɹ	θɜːməkɛɹ	89.00	90.44
prinker	prink	88.64	pɹɪŋkɚ	pɹɪŋk	92.00	90.32
intellicare	intelecare	90.18	ɪntɛlɪkɛɹ	ɪntɛlɨkɛɹ	90.00	90.09
chooey	chooee	88.17	tʃuːi	tʃuːiː	92.00	90.08
dcsl	dcs	90.08	diːsiːɛsɛl	diːsiːɛs	90.00	90.04
mahendra	mahindra	91.08	mæhɛndɹə	mæhɪndɹə	89.00	90.04
lucite	luci	86.67	luːsaɪt	luːsaɪ	93.00	89.83
george	georgine	90.50	dʒɔːɹdʒ	dʒɔːɹdʒɪn	89.00	89.75
tropico	tropicazo	91.78	tɹɑːpɪkoʊ	tɹɑːpɪkɑːzoʊ	87.00	89.39
demiegod	demigods	91.50	dɛmɪeɪgɑːd	dɛmɪgɑːdz	86.00	88.75
mbet	m-bets	85.00	ɛmbɛt	ɛmbɛts	92.00	88.50
fashiongo	fashionego	96.50	fæʃəŋgoʊ	fæʃəniːgoʊ	80.00	88.25
cylance	sylence	75.98	saɪləns	saɪləns	100.00	87.99
ping	pingke	86.67	pɪŋ	pɪŋk	89.00	87.83
pikdare	pi-kare	89.19	pɪkdɛɹ	paɪkɛɹ	86.00	87.60
mbfw	mvfw	80.00	ɛmbiːɛfdʌbəljuː	ɛmviːɛfdʌbəljuː	94.00	87.00
joy	joyme	82.83	dʒɔɪ	dʒɔɪm	91.00	86.92
configon	configo	95.25	kənfɪgən	kənfɪgoʊ	78.00	86.62
prinz	prinse	81.17	pɹɪnts	pɹɪns	92.00	86.58
lovello	lovelle	90.14	lʌvloʊ	lʌvl	83.00	86.57
energeo	enerjo	83.98	ɛnɚdʒeɪoʊ	ɛnɚdʒoʊ	89.00	86.49
trucool	turcool	90.86	tɹuːkuːl	tɜːkuːl	82.00	86.43
carbon	mycarbon	88.83	kɑːɹbən	maɪkɑːɹbən	84.00	86.42
consiglieri	consigliera	93.68	kənsɪglɪɹi	kənsɪgliɛɹə	78.00	85.84
starbucks	charbucks	81.59	stɑːɹbʌks	tʃɑːɹbʌks	90.00	85.80
realme	realmz	88.17	ɹɛlmi	ɹɛlmz	83.00	85.58
axis	traxis	84.44	æksɪs	tɹæksɪs	86.00	85.22
youtube	u-tubes	75.98	juːtuːb	juːtuːbz	94.00	84.99
bimbo	gimbo	83.33	bɪmboʊ	gɪmboʊ	86.00	84.67

tiktok	tiktaktok	85.00	tɪktɑːk	tɪktɐktɑːk	84.00	84.50
z-biome	biome	86.74	ziːbaɪoʊm	baɪoʊm	82.00	84.37
bacchus	cacchus	85.46	bækəs	kækəs	83.00	84.23
philips	philzops	86.07	fɪlɪps	fɪlzəps	80.00	83.04
patter	yatter	85.94	pærɚ	jærɚ	80.00	82.97
noughty	naughtea	73.59	nɔːri	nɔːriə	92.00	82.79
yorxs	yorks	85.33	joːɹksz	jɔːɹks	80.00	82.67
jarlsberg	jørnsberg	82.33	dʒɑːɹlsbɜːg	dʒoːɹnsbɜːg	83.00	82.67
globe-trotter	globetrotter xc	90.23	gloʊbtɹɑːrɚ	gloʊbtɹɑːrɚɹ ɛkssiː	75.00	82.62
treca	trea	92.17	tɹɛkə	tɹiə	73.00	82.58
resolution	resolute	84.75	ɹɛzəluːʃən	ɹɛzəluːt	80.00	82.38
olympéa	olympe	83.98	əlɪmpeɪə	əlɪmp	80.00	81.99
ellesse	elliss	83.22	ɛlɛs	ɛlɪs	80.00	81.61
hugo	hug-o	92.17	hjuːgoʊ	hʌgoʊ	71.00	81.58
initio	vinicio	80.96	ɪnɪɾɪoʊ	vɪnɪsɪoʊ	82.00	81.48
bimbo	bimbolea	84.75	bɪmboʊ	baɪmboʊliə	78.00	81.38
burgerme	burgerly	82.50	bɜːgɚm	bɜːgɚli	80.00	81.25
1link	link	91.17	wʌn lɪŋk	lɪŋk	71.00	81.08
repevax	epvax	86.74	ɹɨpɛvæks	ɛpvæks	75.00	80.87
free	freepour	78.50	fɹiː	fɹiːpɚ	83.00	80.75
zara	zarzar	86.11	zɑːɹɹə	zɑːɹzɑːɹ	75.00	80.56
rabe	rase	80.83	ɹeɪb	ɹeɪz	80.00	80.42
retaron	retlron	89.67	ɹɨtæɹən	ɹɨtlɹɑːn	71.00	80.33
createme	create.	86.07	kɹiːeɪriːm	kɹiːeɪt	74.00	80.04
spa	spato	82.83	spɑː	spɑːroʊ	77.00	79.92
thermomix	termomatrix	84.24	θɜːməmɪks	tɜːməmeɪtɹɪks	75.00	79.62
atma	atmaspa	82.21	ætmə	ætmæspə	77.00	79.61
live	vive	79.17	laɪv	vaɪv	80.00	79.58
cana	canya	92.17	kɑːnə	kænjə	67.00	79.58
l'oreal	joreal	80.96	ɛloːɹiəl	dʒoːɹiəl	78.00	79.48
seiko	seycos	65.50	seɪkoʊ	seɪkoʊz	93.00	79.25
pockit	mypocket	76.47	pɑːkɪt	maɪpɑːkɪt	82.00	79.24
bisleri	bilseri	91.10	baɪslɜːɹi	bɪlsɚɹi	67.00	79.05
kikkoman	kikomand	91.08	kɪkɑːmən	kɪkəmænd	67.00	79.04
fido	fiio	80.83	faɪdoʊ	fɪɪoʊ	77.00	78.92
waken	wakeful	77.21	weɪkən	weɪkfəl	80.00	78.61
nutravita	nootrovita	79.17	nʌtɹevːrə	nuːtɹəviːrə	78.00	78.58
um bongo	ubongo!	84.11	ʌm bɑːŋgoʊ	juːbɑːŋgoʊ	73.00	78.55
pyra	prya	83.75	pɪɹə	pɹaɪə	73.00	78.38
ulma	luma	83.33	ʌlmə	luːmə	73.00	78.17
fransa	fanza	78.50	fɹænsə	fænzə	77.00	77.75
chef	chefchy	82.21	ʃɛf	ʃɛftʃi	73.00	77.61
boss	bossvel	82.21	bɔs	bɔsvəl	73.00	77.61
hanson	hansol	88.17	hænsən	hænsɑːl	67.00	77.58
lucozade	glucos-aid	72.67	luːkəzeɪd	gluːkoʊzeɪd	82.00	77.33
asos	asas	80.83	ɐsoʊz	ɐsæz	73.00	76.92

iqos	niccos	67.50	aɪkoʊz	nɪkoʊz	86.00	76.75
zemo	zoomo	67.11	ziːmoʊ	zuːmoʊ	86.00	76.56
hyprr	hypernft	72.83	haɪpɚ	haɪpɚnft	80.00	76.42
free	freeyoung	75.44	fɹiː	fɹiːjʌŋ	77.00	76.22
bimbo	bimbys	81.17	bɪmboʊ	bɪmbiz	71.00	76.08
uber	youber	84.44	juːbɚ	jaʊbɚ	67.00	75.72
dune	dne	89.25	duːn	diːɛniː	62.00	75.62
scaffeze	scaffx	80.08	skæfɛz	skæfks	71.00	75.54
foltene	foltex	83.98	foʊltiːn	foʊltɛks	67.00	75.49
abanca	abaca	93.56	ɐbæŋkə	æbɑːkə	57.00	75.28
ch	ch_t.	70.50	siːeɪtʃ	siːeɪtʃ tiː	80.00	75.25
suntech	suntank	69.93	sʌntɛk	sʌntæŋk	80.00	74.96
hotpatch	patch	78.92	hɑːtpætʃ	pætʃ	71.00	74.96
huracán	huracanrace	77.53	hjʊɹɹɐkɑːn	hjʊɹɹɐkænɹeɪs	72.00	74.76
free	freetalk	78.50	fɹiː	fɹiːrɔːk	71.00	74.75
free	freeloop	78.50	fɹiː	fɹiːluːp	71.00	74.75
intelect	entelec	77.90	ɪntɛlɨkt	ɛntɛlɛk	71.00	74.45
maplab	maplab.world	78.50	mæplæb	mæplæb wɜːld	70.00	74.25
sacher	sachi	81.17	sæʃɚ	sætʃaɪ	67.00	74.08
fanta	fantarifa	81.06	fæntə	fæntɑːɹɹɪfə	67.00	74.03
fiorelli	fioretto	73.50	fɪoːɹɛli	fɪoːɹɛroʊ	74.00	73.75
sherco	charco	72.39	ʃɜːkoʊ	tʃɑːɹkoʊ	75.00	73.69
vidas	vidya	85.33	viːdəz	vɪdɪə	62.00	73.67
gobox	g-box	84.00	goʊbɑːks	dʒiːbɑːks	63.00	73.50
idee	idee-home	75.44	ɪdiː	ɪdiːhoʊm	71.00	73.22
starbucks	sardarbuksh	76.21	stɑːɹbʌks	sɑːɹdɑːɹbʌkʃ	70.00	73.11
orange	orangery-o-s	78.50	ɔɹɪndʒ	ɔɹɪndʒɚɹioʊɛs	67.00	72.75
free	freeyond	78.50	fɹiː	fɹiːjɑːnd	67.00	72.75
free	freepods	78.50	fɹiː	fɹiːpɑːdz	67.00	72.75
sanytol	savisol	67.07	sænɪtɑːl	sævɪsɑːl	78.00	72.54
snuggledown	snugglemore	81.05	snʌgəldaʊn	snʌgəlmoːɹ	64.00	72.52
pez	pezeeu	77.67	pɛz	pɛziːuː	67.00	72.33
zirco	cozirc	77.61	zɜːkoʊ	kɑːzɜːk	67.00	72.31
glenfiddich	inverfiddich	74.10	glɛnfɪdɪtʃ	ɪnvɜːfɪdɪtʃ	70.00	72.05
salio	saliogen	84.75	sælɪoʊ	sælɪədʒən	59.00	71.88
vallformosa	fermosa	70.77	vælfoːɹmoʊsə	fɜːmoʊsə	73.00	71.88
noughty	nouti	76.17	nɔːri	naʊri	67.00	71.58
tesla	teslapimp	81.06	tɛslə	tɛslɐpɪmp	62.00	71.53
live	life's	70.00	laɪv	laɪfz	73.00	71.50
e-bulli	bullit	80.96	iːbʊli	bʊlɪt	62.00	71.48
bimbo	bims	75.92	bɪmboʊ	bɪmz	67.00	71.46
genie	genai	85.33	dʒiːni	dʒɛnaɪ	57.00	71.17
lakme	like-me	70.32	lækmi	laɪkmiː	71.00	70.66
kelio	kleeo	70.25	kɛlɪoʊ	kliːoʊ	71.00	70.62
terry	terrissa	74.00	tɛɹi	tɛɹɪsə	67.00	70.50
tygrys	tigris	73.50	tɪgɹiz	taɪgɹɪs	67.00	70.25

nike	nuke	80.00	naɪk	nu:k	60.00	70.00
007	skx007	58.50	ziəɹoʊziəɹoʊ sɛvən	ɛskeɪɛks ziəɹoʊziəɹoʊ sɛvən	81.00	69.75
geneverse	genv3rse	85.28	dʒɛnɪvɜ:s	dʒɛnv θɹi: ɑ:ɹɹɛsi:	53.00	69.14
lego	solego	76.11	lɛgoʊ	sɑ:li:goʊ	62.00	69.06
perry	perryhome	81.06	pɛɹi	pɛɹɪhoʊm	57.00	69.03
kadawe	kademae	80.89	kædɔ:	keɪdmi:	57.00	68.94
acutil	accudis	70.84	ɐkju:rɪl	ɐkju:diz	67.00	68.92
bru	bruys	82.83	bɹu:	bɹaɪz	55.00	68.92
bimbo	wimko	66.67	bɪmboʊ	wɪmkoʊ	71.00	68.83
cazoo	carkoo	79.39	kæzu:	kɑ:ɹku:	57.00	68.19
doctolib	avocatlib	75.78	dɑ:ktəlɪb	ævəkætlɪb	60.00	67.89
boss	kissboss	62.67	bɔs	kɪsbɔs	73.00	67.83
bmw	bmv	74.61	bi:ɛmdʌbəlju:	bi:ɛmvi:	61.00	67.81
marca	plusmarca	57.35	mɑ:ɹkə	plʌsmɑ:ɹkə	78.00	67.68
mdh	mhs	61.28	ɛmdi:eɪtʃ	ɛmeɪtʃɛs	74.00	67.64
align	clickalign	60.17	ɐlaɪn	klɪkɐlaɪn	75.00	67.58
ajona	avoma	68.00	ædʒoʊnə	ævoʊmə	67.00	67.50
zara	zaraphora	75.44	zɑ:ɹɹə	zæɹɐfo:ɹə	59.00	67.22
levi's	levigo	76.83	lɛviz	lɛvɪgoʊ	57.00	66.92
zara	zareus	71.25	zɑ:ɹɹə	zɛɹəs	62.00	66.62
zara	zareus	71.25	zɑ:ɹɹə	zɛɹəs	62.00	66.62
naturli'	natureal	82.50	neɪrɜ:li	neɪtʃɚɹiəl	50.00	66.25
moncler	northcler	70.29	mɔŋklɚ	nɔ:ɹθklɚ	62.00	66.14
airbnb	airbrick	70.17	ɛɹbnb	ɛɹbɹɪk	62.00	66.08
resolva	consolva	69.15	ɹɨzɑ:lvə	kənsɑ:lvə	63.00	66.08
sanytol	sanatio	78.83	sænɪtɑ:l	sæneɪʃɪoʊ	53.00	65.92
moncler	montec	73.39	mɔŋklɚ	mɔntɛk	57.00	65.19
apiretal	a'peal	77.38	ɐpaɪɚrəl	ɐpi:l	53.00	65.19
very	veryco	86.67	vɛɹi	vɜ:ɹɪkoʊ	43.00	64.83
bimbo	vibo	72.67	bɪmboʊ	vi:boʊ	57.00	64.83
head	headoniste	72.50	hɛd	hɛdəni:st	57.00	64.75
saypha	shaype	73.50	seɪfə	ʃeɪp	55.00	64.25
helios	delio	77.61	hɛlɪoʊz	dɨli:oʊ	50.00	63.81
coversyl	covixyl-v	69.94	kʌvɚsɪl	kɑ:vɪksɪlvi:	57.00	63.47
simoniz	permanize	58.60	sɪmənɪz	pɜ:mənaɪz	67.00	62.80
vfh	vfhonline	67.22	vi:ɛfeɪtʃ	vi:ɛfhɑ:nlaɪn	58.00	62.61
rolex	dermarollex	49.03	ɹoʊlɛks	dɜ:mɚɹoʊlɛks	76.00	62.52
apple	alpineapple	62.89	æpəl	ælpɪni:pəl	62.00	62.45
thermomix	zaubermix	63.19	θɜ:məmɪks	zɔ:bɚmɪks	60.00	61.59
magnavox	multivox	58.33	mægnɐvɑ:ks	mʌltivɑ:ks	64.00	61.17
nutella	mixitella	68.83	nu:tɛlə	mɪksaɪtɛlə	53.00	60.92
airbnb	francebnb	59.65	ɛɹbnb	fɹænsɛbnb	62.00	60.82
curve	crv	81.50	kɜ:v	si:ɑ:ɹvi:	40.00	60.75
gallo	rampingallo	52.52	gæloʊ	ɹæmpɪŋgæloʊ	67.00	59.76
iphone	mifon	62.50	aɪfoʊn	mɪfɑ:n	57.00	59.75

joy	bjoie	59.44	dʒɔɪ	bjɔɪ	60.00	59.72
jd	jdyaoying	57.63	dʒeɪdiː	dʒeɪdaɪeɪɑːiɪŋ	61.00	59.31
bally	ballyclare	78.50	bɔːli	bælɪklɛɹ	40.00	59.25
swift	microswift	55.17	swɪft	maɪkɹoʊswɪft	63.00	59.08
bloo	bluuwash	45.67	bluː	bluːwɑːʃ	71.00	58.33
head	superhead	53.69	hɛd	suːpɚhɛd	62.00	57.84
trek	gotrekfeel	68.50	tɹɛk	gɑːtɹɪkfiːl	47.00	57.75
blippi	bbibbi	58.33	blɪpi	biːbɪbi	57.00	57.67
immun44	immuno-19	73.70	ɪmʌn foːɹri foːɹ	ɪmjuːnoʊ naɪntiːn	40.00	56.85
rolex	relxhome	57.17	ɹoʊlɛks	ɹɨlkshoʊm	56.00	56.58
kpn	opn	72.39	keɪpiːɛn	ɑːpən	40.00	56.19
mc	macbeans	58.75	ɛmsiː	məkbiːnz	53.00	55.88
ape	apecessories	61.25	eɪp	eɪpɪsɛsɚɹiz	50.00	55.62
airbnb	marseillebnb	57.17	ɛɹbnb	mɑːɹseɪlɛbnb	53.00	55.08
facebook	motherbook	60.08	feɪsbʊk	mʌðɚbʊk	50.00	55.04
alaïa	azzaia	64.00	ɐlæiːə	æzeɪə	46.00	55.00
puma	coma	58.33	puːmə	koʊmə	50.00	54.17
bimbo	amorbimbi	55.17	bɪmboʊ	ɐmoːɹbɪmbaɪ	53.00	54.08
azure	azurity	77.21	æʒɚ	æzjʊɹɹɨri	29.00	53.11
bimbo	binbokplay	65.83	bɪmboʊ	baɪnbɑːkpleɪ	40.00	52.92
zara	zorazone	54.86	zɑːɹɹə	zoːɹɐzoʊn	47.00	50.93
matters	m4tter	81.71	mærɚz	ɛm foːɹ tiːtɜː	19.00	50.36
quirón	quiromasté	59.44	kwɜːɹɑːn	kwɪɹəmɐsteɪ	38.00	48.72
joy	joïsta	55.33	dʒɔɪ	dʒɑːiːstə	40.00	47.67
louboutin	lubov	61.74	laʊbaʊtɪn	luːbɑːv	33.00	47.37
we	wecotton	60.00	wiː	wɛkəʔn̩	33.00	46.50
mcdonalds	mcsweet	44.13	məkdɑːnəldz	məkswiːt	48.00	46.07
md	intimd	25.00	ɛmdiː	ɪntɪmdiː	67.00	46.00
sane	cbdsane	36.50	seɪn	siːbiːdiːseɪn	53.00	44.75
book	restaubook	28.50	bʊk	ɹɨstaʊbʊk	57.00	42.75
h10	motel 10	18.00	eɪtʃ tɛn	moʊtɛl tɛn	60.00	39.00
coco	kokomarina	42.83	koʊkoʊ	kɑːkəmɚɹiːnə	30.00	36.42
mi	lovmi	28.50	maɪ	lʌvmi	40.00	34.25